Kementerian Pendidikan dan Kebudayaan
Republik Indonesia
2013



Pemrograman Web 2

UNTUK SMK / MAK KELAS X



Pemrograman Web

Penulis : **Kadek Surya P., Endah Damayanti**

Editor Materi : **Wahyu P.**

Editor Bahasa : **Wahyu P.**

Ilustrasi Sampul :

Desain & Ilustrasi Buku : **PPPPTK BOE Malang**

Pusat Pengembangan & Pemberdayaan Pendidik & Tenaga Kependidikan Bidang Otomotif & Elektronika:

Jl. Teluk Mandar, Arjosari Tromol Pos 5, Malang 65102, Telp. (0341) 491239, (0341) 495849, Fax. (0341) 491342, Surel: vedcmalang@vedcmalang.or.id, Laman: www.vedcmalang.com

DISKLAIMER (DISCLAIMER)

Penerbit tidak menjamin kebenaran dan keakuratan isi/informasi yang tertulis di dalam buku tek ini. Kebenaran dan keakuratan isi/informasi merupakan tanggung jawab dan wewenang dari penulis.

Penerbit tidak bertanggung jawab dan tidak melayani terhadap semua komentar apapun yang ada didalam buku teks ini. Setiap komentar yang tercantum untuk tujuan perbaikan isi adalah tanggung jawab dari masing-masing penulis.

Setiap kutipan yang ada di dalam buku teks akan dicantumkan sumbernya dan penerbit tidak bertanggung jawab terhadap isi dari kutipan tersebut. Kebenaran keakuratan isi kutipan tetap menjadi tanggung jawab dan hak diberikan pada penulis dan pemilik asli. Penulis bertanggung jawab penuh terhadap setiap perawatan (perbaikan) dalam menyusun informasi dan bahan dalam buku teks ini.

Penerbit tidak bertanggung jawab atas kerugian, kerusakan atau ketidaknyamanan yang disebabkan sebagai akibat dari ketidakjelasan, ketidaktepatan atau kesalahan didalam menyusun makna kalimat didalam buku teks ini.

Kewenangan Penerbit hanya sebatas memindahkan atau menerbitkan mempublikasi, mencetak, memegang dan memproses data sesuai dengan undang-undang yang berkaitan dengan perlindungan data.

Katalog Dalam Terbitan (KDT)

Teknik Komputer dan JaringanEdisi Pertama 2013

Kementerian Pendidikan & Kebudayaan

Direktorat Jenderal Peningkatan Mutu Pendidik & Tenaga Kependidikan, th. 2013: Jakarta

KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan yang Maha Esa atas tersusunnya buku teks ini, dengan harapan dapat digunakan sebagai buku teks untuk siswa Sekolah Menengah Kejuruan (SMK) Bidang Studi Keahlian Teknologi InformasidanKomunikasi,Program Keahlian TeknikKomputerdanIJaringan.

Penerapan kurikulum 2013 mengacu pada paradigma belajar kurikulum abad 21 menyebabkan terjadinya perubahan, yakni dari pengajaran (*teaching*) menjadi BELAJAR (*learning*), dari pembelajaran yang berpusat kepada guru (*teachers-centered*) menjadi pembelajaran yang berpusat kepada peserta didik (*student-centered*), dari pembelajaran pasif (*pasive learning*) ke cara belajar peserta didik aktif (*active learning-CBSA*) atau *Student Active Learning-SAL*.

Buku teks ″Pemrograman Web″ini disusun berdasarkan tuntutan paradigma pengajaran dan pembelajaran kurikulum 2013diselaraskan berdasarkan pendekatan model pembelajaran yang sesuai dengan kebutuhan belajar kurikulum abad 21, yaitu pendekatan model pembelajaran berbasis peningkatan keterampilan proses sains.

Penyajian buku teks untuk Mata Pelajaran ″Pemrograman Web”ini disusun dengan tujuan agar supaya peserta didik dapat melakukan proses pencarian pengetahuan berkenaan dengan materi pelajaran melalui berbagai aktivitas proses sains sebagaimana dilakukan oleh para ilmuwan dalam melakukan eksperimen ilmiah (penerapan scientifik), dengan demikian peserta didik diarahkan untuk menemukan sendiri berbagai fakta, membangun konsep, dan nilai-nilai baru secara mandiri.

Kementerian Pendidikan dan Kebudayaan, Direktorat PembinaanSekolah Menengah Kejuruan, dan Direktorat Jenderal Peningkatan Mutu Pendidik dan Tenaga Kependidikan menyampaikan terima kasih, sekaligus saran kritik demi kesempurnaan buku teks ini dan penghargaan kepada semua pihak yang telah berperan serta dalam membantu terselesaikannya buku teks siswa untuk Mata Pelajaran ″Pemrograman Web”kelas X/Semester 2Sekolah Menengah Kejuruan (SMK).

Jakarta, 12 Desember 2013
Menteri Pendidikan dan Kebudayaan

Prof. Dr. Mohammad Nuh, DEA

DAFTAR ISI

GLOSARIUM

ActionScript: bahasa pemrograman mirip Javascript pada aplikasi Adobe Flash.

Array asosiatif: jenis array yang menggunakan nama sebagai pengganti index untuk menunjuk data yang disimpan.

Array: himpunan data sejenis yang disimpan dalam suatu variabel dengan index untuk mengakses setiap data yang tersimpan.

Asynchronous Javascript and XML (AJAX): suatu teknologi yang dikembangkan untuk memungkinkan suatu proses berjalan secara asinkron dengan proses lainnya (multi-proses) dalam satu halaman web.

Autocomplete: kemampuan suatu komponen melakukan pencarian dari data yang ada berdasarkan input dari user.

Autofocus: atribut pengaturan fokus pada suatu elemen setelah halaman di load.

Boolean: tipe untuk mewakili data dengan format benar salah.

Border: batas yang mengelilingi padding dan content.

Cascading Style Sheet (CSS): standar yang digunakan untuk mengatur tampilan suatu halaman HTML

Case Sensitive: bentuk pernyataan yang akan membedakan antara huruf besar dan kecil untuk nama yang sama.

Cols: jumlah kolom karakter yang dapat ditampilkan oleh textarea.

Disabled: atribut untuk menonaktifkan mode entri pada suatu komponen formulir HTML.

ECMAScript: Nama lain untuk menyatakan standar Javascript.

Embedded style sheet: CSS yang ditempatkan dalam tag <style>.

Event: sebutan dari satu aksi yang dilakukan oleh user atau dapat juga berasal dari browser untuk diolah oleh aplikasi.

External style sheet: CSS yang ditempatkan pada file tersendiri, umumnya file dengan ekstensi *.css.

Fungsi: kumpulan program untuk menjalankan suatu proses tertentu.

Inline style sheet: CSS yang ditempatkan pada suatu elemen HTML.

Java: bahasa pemrograman untuk pengembangan aplikasi lintas platform menggunakan mesin virtual Java.

Javascript: bahasa pemrograman untuk pengolahan halaman HTML

Komentar program: bagian dari program yang tidak akan dieksekusi interpreter ataupun compiler.

Komponen checkbox: komponen formulir HTML untuk menentukan beberapa pilihan dari yang ada.

Komponen datalist: komponen formulir HTML untuk menampilkan data lebih dari dengan dukungan autocomplete.

Komponen input file: komponen formulir HTML untuk pengiriman file.

Komponen input image: komponen formulir HTML untuk membuat tombol dari gambar.

Komponen radio button: komponen formulir HTML untuk memilih satu dari sekian pilihan.

Komponen select: komponen formulir HTML untuk menampilkan lebih dari satu pilihan.

Margin: daerah paling diluar yang membatasi dengan elemen HTML yang lainnya dan tidak berwarna.

Multi-purpose Internet Mail Extensions (MIME): perluasan standar email, sekarang juga digunakan secara meluas untuk standar lainnya di internet, untuk mendukung pengiriman data dengan format teks selain ASCII, non-teks dan penerapan banyak format dalam satu file (multipart message bodies).

Notasi camel case: notasi penamaan yang menggunakan huruf kapital pada setiap kata-nya tanpa mengandung spasi.

Notasi hungaria: notasi penamaan yang memberikan awalan berupa jenis data yang akan ditampung oleh suatu variabel.

Padding: area yang mengelilingi content.

Pemrograman berorientasi obyek: pemrograman yang menggambarkan suatu aplikasi sebagai interaksi antar obyek seperti yang terjadi dalam kehidupan sehari-hari.

Property: bagian CSS untuk menunjuk detail modifikasi yang akan dilakukan pada suatu elemen HTML.

Pustaka: kumpulan konstanta, fungsi, kelas atau obyek dalam pemrograman.

Rows: jumlah baris teks yang dapat ditampilkan oleh textarea.

Selector: elemen HTML, yang ditunjuk melalui atribut class, ID atau nama tag, yang akan dimodifikasi tampilannya oleh CSS.

String: tipe untuk mewakili format data yang berupa kumpulan karakter.

Struktur percabangan: struktur program yang memiliki kondisi tertentu untuk menjalankan alur yang berbeda.

Struktur perulangan: struktur program yang memungkinkan suatu kumpulan perintah dijalankan berkali-kali dengan terpenuhinya suatu kondisi.

Textarea: komponen formulir HTML untuk menampung teks yang panjang.

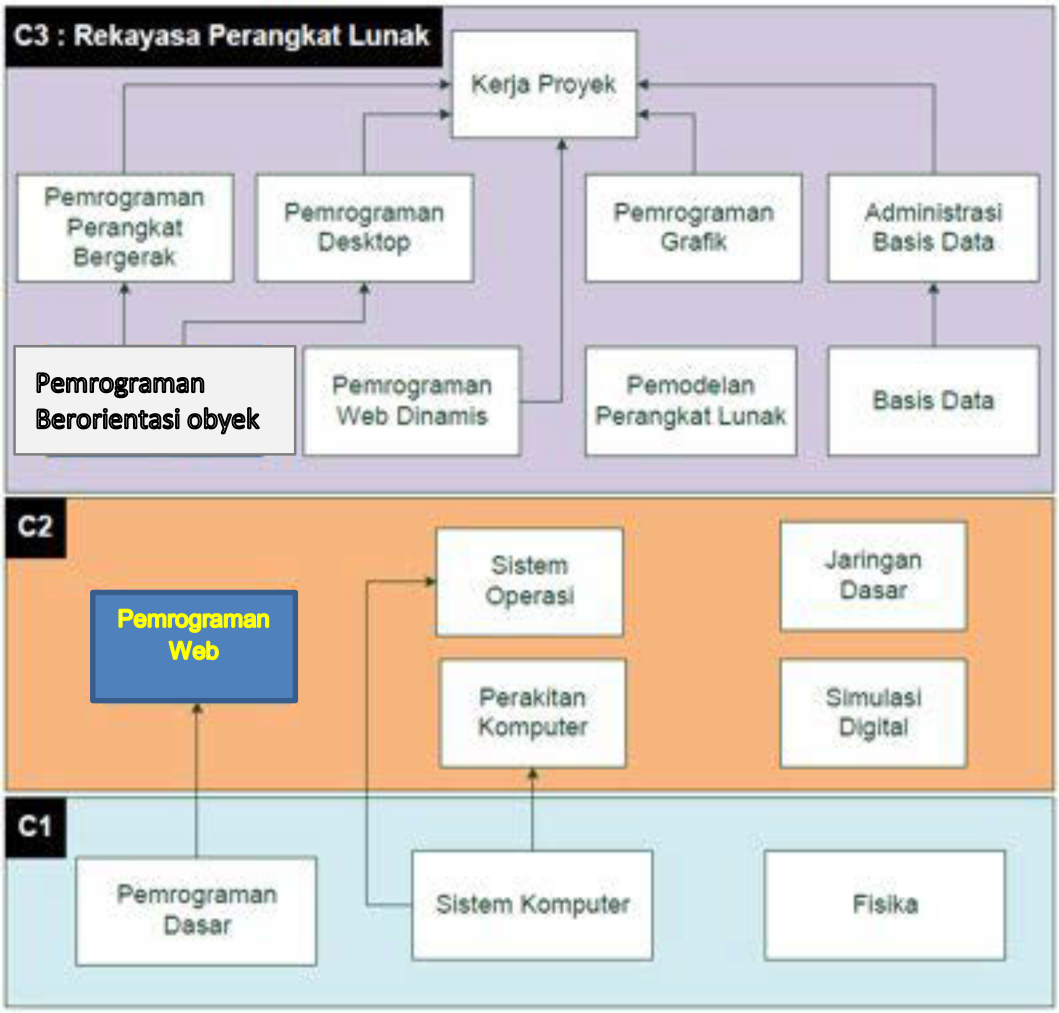
World Wide Web Consortium (W3C): organisasi internasional yang mengatur standar aplikasi yang berhubungan dengan teknologi web.

Wrap: atribut pengemasan teks pada komponen textarea.



PETA KEDUDUKAN BAHAN AJAR

Peta kedudukan bahan ajar merupakan suatu diagram yang menjelaskan struktur mata pelajaran dan keterkaitan antar mata pelajaran dalam satu kelompok bidang paket keahlian. Gambar 1 menjelaskan peta kedudukan bahan ajar untuk paket keahlian rekayasa perangkat lunak . Gambar 1 menjelaskan peta kedudukan bahan ajar untuk paket keahlian rekayasa perangkat lunak . Kelompok C1 merupakan kelompok mata pelajaran wajib dasar bidang kejuruan. C2 merupakan kelompok mata pelajaran wajib dasar kompetensi kejuruan dan C3 merupakan kelompok mata pelajaran wajib paket keahlian.



Gambar 1. Peta Kedudukan Bahan Ajar Kelompok C2
Mata Pelajaran Pemrograman Web

Sementara itu peta konsep mata pelajaran menjelaskan struktur urutan kegiatan belajar dan topik materi pelajaran. Gambar 2 dibawah ini menjelaskan peta konsep mata pelajaran Pemrograman web untuk kelas X semester 2.



Gambar 2. Peta Konsep Mata Pelajaran Pemrograman Web Kelas X Semester 2



BAB 1 PENDAHULUAN

A.Diskripsi

Pemrograman web 2 merupakan salah satu mata pelajaran wajib dasar pada dasar program keahlian Teknik Komputer dan Informatika (TKI). Berdasarkan struktur kurikulum mata pelajaran Pemrograman web 2 disampaikan dikelas X semester 2 yang disampaikan dalam waktu 4 jam pelajaran per minggu.

Pada semester 2 ini materi pemrograman web ditekankan pada perintah-perintah lanjut pada HTML untuk pembuatan halaman dan perintah – perintah menggunakan java script. Perintah HTML yang diajarkan pada pemrograman web 2 ini meliputi pembuatan komponen formulir serta pemberian style pada suatu halaman web.

Untuk materi java script meliputi teknik pemrograman halaman web , pengolahan input user. Pada teknik pemrograman halaman web akan dijelaskan lebih lanjut tentang anatomi dan cara kerja kode javascript, dasar pemrograman client (variabel, tipe data, operator), array dimensi 1 dan multidimensi, struktur kontrol percabangan pada program client, struktur kontrol perulangan pada program client, fungsi bawaan dan buatan user pada program client.

B.Prasyarat

Untuk kelancaran pencapaian kompetensi dalam mata pelajaran pemrograman web ini dibutuhkan beberapa peryaratan baik pengetahuan maupun ketrampilan dasar. Persyaratan tersebut antara lain ialah: peserta didik mempunyai kompetensi dalam hal pemanfaatan teknologi informasi, seperti mengoperasikan hardware komputer dan mengoperasikan perangkat lunak aplikasi. Perangkat lunak aplikasi tersebut antar lain ialah pengolah data untuk menganalisis data hasil eksperimen, pengolah kata untuk membuat laporan dan aplikasi presentasi untuk mengkomunikasikan dan mempresentasikan hasil laporan.

C.Petunjuk

Buku pedoman siswa ini disusun berdasarkan kurikulum 2013 yang mempunyai ciri khas penggunaan metode scientific. Buku ini terdiri dari dua bab yaitu bab satu pendahuluan dan bab dua pembelajaran. Dalam bab pendahuluan beberapa yang harus dipelajari peserta didik adalah diskripsi mata pelajaran yang berisi informasi umum, rasionalisasi dan penggunaan metode scientifik. Selanjutnya pengetahuan tentang persyaratan, tujuan yang diharapkan, kompetensi inti dan dasar yang akan dicapai serta test kemampuan awal.

Bab dua menuntun peserta didik untuk memahami deskripsi umum tentang topik yang akan dipelajari dan rincian kegiatan belajar sesuai dengan kompetensi dan tujuan yang akan dicapai. Setiap kegiatan belajar terdiri dari tujuan dan uraian materi topik pembelajaran, tugas serta test formatif. Uraian pembelajaran berisi tentang diskripsi pemahaman topik materi untuk memenuhi kompetensi pengetahuan. Uraian pembelajaran juga menjelaskan deskripsi unjuk kerja atau langkah-langkah logis untuk memenuhi kompetensi skill.

Tugas yang harus dikerjakan oleh peserta didik dapat berupa tugas praktek, eksperimen atau pendalaman materi pembelajaran. Setiap tugas yang dilakukan melalui beberapa tahapan scientifik yaitu : 1) melakukan pengamatan setiap tahapan unjuk kerja 2) melakukan praktek sesuai dengan unjuk kerja 3) mengumpulkan data yang dihasilkan setiap tahapan 4) menganalisa hasil data menggunakan analisa diskriptif 5) mengasosiasikan beberapa pengetahuan dalam uraian materi pembelajaran untuk membentuk suatu kesimpulan 6) mengkomunikasikan hasil dengan membuat laporan portofolio. Laporan tersebut merupakan tagihan yang akan dijadikan sebagai salah satu referensi penilaian

D.Tujuan Akhir

Setelah mempelajari uraian materi dalam bab pembelajaran dan kegiatan belajar diharapkan peserta didik dapat memiliki kompetensi sikap, pengetahuan dan ketrampilan yang berkaitan dengan materi:

- Format formulir halaman web
- Style halaman web
- Teknik pemrograman web
- Akses Komponen web

E.Kompetensi inti dan kompetensi dasar

1. Kompetensi Inti 1 : Menghayati dan mengamalkan ajaran agama yang dianutnya.

   Kompetensi Dasar :

   1.1. Memahami nilai-nilai keimanan dengan menyadari hubungan keteraturan dan kompleksitas alam dan jagad raya terhadap kebesaran Tuhan yang menciptakannya

   1.2. Mendeskripsikan kebesaran Tuhan yang menciptakan berbagai sumber energi di alam

   1.3. Mengamalkan nilai-nilai keimanan sesuai dengan ajaran agama dalam kehidupan sehari-hari.

2. Kompetensi Inti 2: Menghayati dan Mengamalkan perilaku jujur, disiplin, tanggung jawab, peduli (gotong royong, kerjasama, toleran, damai), santun, responsif dan proaktif dan menunjukan sikap sebagai bagian dari solusi atas berbagai permasalahan dalam berinteraksi secara efektif dengan lingkungan sosial dan alam serta dalam menempatkan diri sebagai cerminan bangsa dalam menempatkan diri sebagai cerminan bangsa dalam pergaulan dunia.

   Kompetensi Dasar:

   2.1. Menunjukkan perilaku ilmiah (memiliki rasa ingin tahu; objektif; jujur; teliti; cermat; tekun; hati-hati; bertanggung jawab; terbuka; kritis; kreatif; inovatif dan peduli lingkungan) dalam aktivitas sehari-hari sebagai wujud implementasi sikap dalam melakukan percobaan dan berdiskusi

   2.2. Menghargai kerja individu dan kelompok dalam aktivitas sehari-hari sebagai wujud implementasi melaksanakan percobaan dan melaporkan hasil percobaan.

3. Kompetensi Inti 3: Memahami,menerapkan dan menganalisis pengetahuan faktual, konseptual dan prosedural berdasarkan rasa ingintahunya tentang ilmu pengetahuan, teknologi, seni, budaya, dan humaniora dalam wawasan kemanusiaan, kebangsaan, kenegaraan, dan peradaban terkait penyebab fenomena dan kejadian dalam bidang kerja yang spesifik untuk memecahkan masalah.

   Kompetensi Dasar:

3.6. Memahami format formulir pada halaman web

3.7. Memahami style pada halaman web

3.8. Memahami teknik pemrograman pada halaman web

4. Kompetensi Inti 4: Mengolah, menalar, dan menyaji dalam ranah konkret dan ranah abstrak terkait dengan pengembangan dari yang dipelajarinya di sekolah secara mandiri, dan mampu melaksanakan tugas spesifik dibawah pengawasan langsung.

   Kompetensi Dasar:

   4.6. Menyajikan formulir pada halaman web

   4.7. Menyajikan style tertentu pada halaman web

   4.8. Menyajikan teknik-teknik dalam pemrograman web

BAB 2 PEMBELAJARAN

A. Diskripsi

Pemrograman web 2 merupakan salah satu mata pelajaran wajib dasar pada dasar program keahlian Teknik Komputer dan Informatika (TKI). Berdasarkan struktur kurikulum mata pelajaran Pemrograman web 2 disampaikan dikelas X semester 2 yang disampaikan dalam waktu 4 jam pelajaran per minggu.

Pada semester 2 ini materi pemrograman web ditekankan pada perintah-perintah lanjut pada HTML untuk pembuatan halaman dan perintah – perintah menggunakan java script. Perintah HTML yang diajarkan pada pemrograman web 2 ini meliputi pembuatan komponen formulir serta pemberian style pada suatu halaman web.

Untuk materi java script meliputi teknik pemrograman halaman web , pengolahan input user. Pada teknik pemrograman halaman web akan dijelaskan lebih lanjut tentang anatomi dan cara kerja kode javascript, dasar pemrograman client (variabel, tipe data, operator), array dimensi 1 dan multidimensi, struktur kontrol percabangan pada program client, struktur kontrol perulangan pada program client, fungsi bawaan dan buatan user pada program client.

B. Kegiatan Belajar

1. Kegiatan Belajar 1 : Menyajikan Komponen Entri Teks

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 1 ini siswa diharapkan dapat :

1) Memahami komponen entri teks
2) Menyajikan komponen entri teks ke dalam sebuah halaman web

b. Uraian materi.

Form HTML digunakan untuk memilih pelbagai macam inputan yang disediakan. Data bisa diinputan dalam bentuk textarea, input teks , pilihan radio button, check box dan lain sebagainya. Bentuk inputan tersebut dikemas elemen dalam tag-tag form. Pada kegiatan belajar ini akan mempelajari inputan text.

1) Komponen text area multiline

Komponen text area merupakan area tempat untuk menyimpan text atau tulisan baru, atau bisa dikatakan sebuah komponen HTML yang digunakan untuk menampilkan text dalam bentuk dan format text.

Untuk menampung input teks yang panjang dan mungkin tersusun atas lebih dari 1 baris, maka digunakan komponen input ‘textarea’. TEXTAREA dapat digabungkan dengan tag FORM untuk menjadi inputan yang akan diolah oleh file pemroses sebagaimana seperti yang telah dijelaskan pada tag FORM tersebut.

Format umum penulisan text area multiline

```
<textarea  atribute=”atribute">
```

Terdapat beberapa atribut dari text area, diantaranya row,col dan lain sebagainya. Pada HTML 5 menyediakan banyak 6 atribut baru textarea.

Atribut	Value	Keterangan
Autofocus	autofocus	Area tertentu pada text area, dimana

→baru dihtml 5		kursor y otomatis mengarah ke daerah tersebut ketika halaman web diload.
Cols	number	Menentukan lebar text area
Disabled	disabled	Text area dapat diubah
Form	form_id	Satu atau lebih form pada text area
→baru dihtml 5		Menentukan panjang karakter pada text area
Maxlength →baru dihtml 5	Number	Menentukan panjang karakter pada text area
Name	text	Nama dari text area
Placeholder →baru dihtml 5	text	Memberikan gambaran singkat tentang nilai pada tex tarea
Readonly	readonly	Text area bersifat read-only
Required →baru dihtml 5	required	Menentukan text area tersebut harus diisi atau tidak.
Rows	number	Menentukan tinggi text area
Wrap →baru dihtml 5	hard soft	Menentukan bagaimana cara teks dikemas dalam text area sebelum teks tersebut dikirimkan

NAME

Atribut ini digunakan untuk meemberikan nama dari TEXTAREA. Atribut ini akan sangat berguna jika tag TEXTAREA digabungkan dengan tag FORM.

NAME

Atribut ini digunakan untuk meemberikan nama dari TEXTAREA. Atribut ini akan sangat berguna jika tag TEXTAREA digabungkan dengan tag FORM.

Contoh penulisan :

```
<!DOCTYPE html>
<html>
<head>
<title>cek form </title>
</head>
<body>
<p> Silahkan diisi pada teks area yang tersedia</p>
<p><textarea NAME="papantulis" COLS=40 ROWS=6>
</textarea></p>
</body>
</html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan pada halaman web seperti berikut :



Gambar 1.1 penerapan textarea dengan atribut name

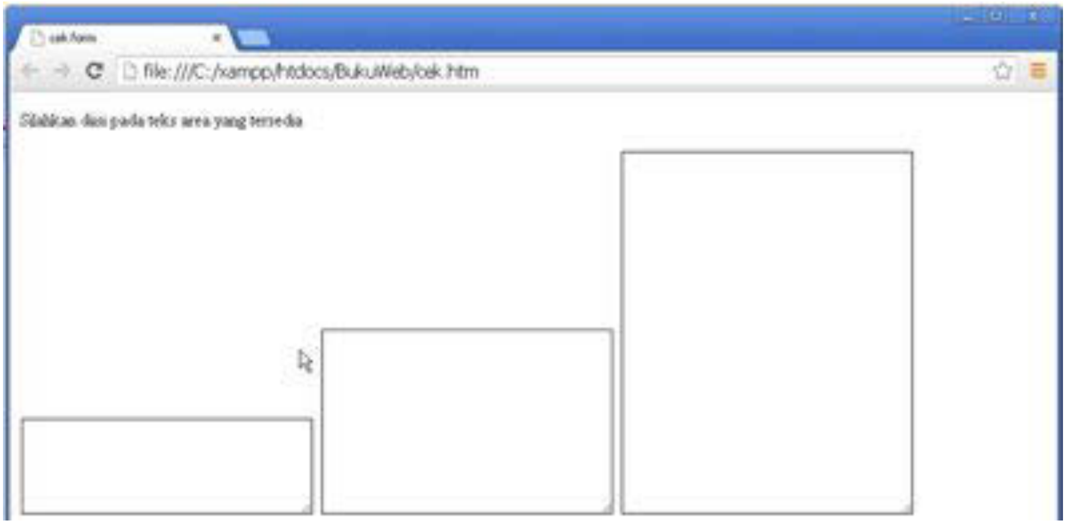
COLS

Atribut COLS digunakan untuk menentukan lebar dari TEXTAREA. Contoh berikut menampilkan 3 komponen textarea dengan lebar textarea bervariasi :

```
<!DOCTYPE html>
<html>
<head>
<title>cek form </title>
</head>
<body>
```

```
<p> Silahkan diisi pada teks area yang tersedia</p>
<TEXTAREA NAME="rendah" COLS=30 ROWS=5></TEXTAREA>
<TEXTAREA NAME="sedang" COLS=30 ROWS=10></TEXTAREA>
<TEXTAREA NAME="tinggi" COLS=30 ROWS=20></TEXTAREA>
</body>
</html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan pada halaman web seperti berikut :



Gambar 1.2 penerapan textarea dengan atribut rows

READONLY

Atribut ini digunakan agar tulisan yang terdapat didalam TEXT AREA hanya dapat dibaca dan disalin tapi tidak dapat diubah.

Contoh :

```
<!DOCTYPE html>
<html>
<head>
<title>cek form </title>
</head>
<body>
<TEXTAREA NAME="untukdibaca" COLS=40 ROWS=6 READONLY>
Tulisan ini tidak dapat diubah tapi bisa disalin.
</TEXTAREA>
```

```
</body>
</html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan pada halaman web seperti gambar dibawah ini. Bila kursor diarahkan ke textarea maka kursor tidak bisa aktif dan tidak bisa mengedit tulisan yang berada dalam textarea, hanya bisa dibaca saja.



Gambar 1.3 penerapan textarea dengan atribut readonly

DISABLED

Atribut ini digunakan agar tulisan yang terdapat didalam TEXTAREA tidak dapat disalin dan diubah.

Contoh :

```
<!DOCTYPE html>
<html>
<head>
<title>cek form </title>
</head>
<body>
<TEXTAREA NAME="terlarang" COLS=40 ROWS=6 DISABLED>
 Tulisan ini tidak dapat disalin maupun diubah.
</TEXTAREA>
</body>
</html>
```

Maka hasilnya akan terlihat seperti berikut ini,



Gambar 1.4 penerapan textarea dengan atribut disabled

2) **Komponen input text password**

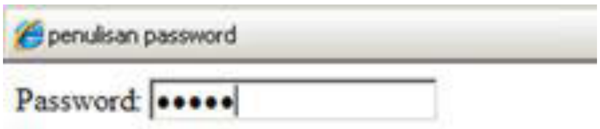
Komponen input text password bertanggung jawab untuk memasukan data password. Dari atribut-atribut tersebut, yang utama dan terutama harus disesuaikan atau diberi nama sesuai dengan item datanya adalah atribut ‘name’, karena file yang dituju pada atribut action di tag form, yang biasanya adalah file server side scripting (PHP) akan mengambil input dari form berdasarkan atribut nama (‘name’) dari komponen input form tersebut.Disamping tipe text, terdapat pula tipe input teks yang lain, yakni ‘password’. Input tpe ‘password’ ini akan menghasilkan input dengan notasi ’●’. Biasanya input tipe ‘password’ digunakan dalam form untuk masukan kata kunci atau password dari pengguna di dalam form login.Format dari elemen input text password HTML-nya adalah sebagai berikut :

```
<input type=password name=name>
<input type=password name=name maxlength=length>
<input type=password name=name size=size>
<input type=password name=name value=value>
```

Berikut adalah penggalan listing program penulisan form input password, dimana form diberi nama=pwd dengan panjang maximal karakternya adalah 6.

```
<!doctype html>
<html>
<head>
<title>penulisan password </title>
</head>
<body>
<form >
Password: <input type="password" name="pwd" maxlength="6">
</form>
</body>
</html>
```

Contoh hasil kompilasi dari listing diatas :



Gambar 1.5 form input password

3) Komponen input text

Komponen input text merupakan komponen untuk memasukan data text ke server dalam bentuk textfield.Format tag HTML-nya adalah sebagai berikut :

```
<INPUT TYPE=TEXT NAME=name>
<INPUT TYPE=TEXT NAME=name MAXLENGTH=length>
<INPUT TYPE=TEXT NAME=name SIZE=size>
<INPUT TYPE=TEXT NAME=name VALUE=value>
```

Berikut contoh listing program yang menggunakan form input
Contoh bentuk tampilannya adalah sebagai berikut :

```
<!doctype html>
<html>
<head>
<title>input text</title>
```

```
</head>
<body>
<form>
Nama : <input type="text" name="nama"><br>
Sekolah : <input type="text" name="sekolah">
</form>
</body>
</html>
```



Gambar 1.6 form input text

4) Form menggunakan input hidden

Format tag form input hidden HTML-nya adalah sebagai berikut :

```
<INPUT TYPE=HIDDEN NAME=name VALUE=value>
```

Pada format form input hidden ditandai dengan atribut form type =hidden pada tag <input>, yang artinya ada form inputan yang disembunyikan atau tidak tidak tampilkan. Untuk selengkapnya dapat dilihat pada program berikut

Bila listing program diatas dijalankan dibrowser akan menghasilkan tampilan sebagai berikut :

```
<!doctype html>
<html>
<head>
<title>hidden input file </title>
</head>
<body>
<form >
  Nama : <input type=” name=”fnama”><br>
<input type=”hidden” name=”negara” value=”Indonesia”>
```

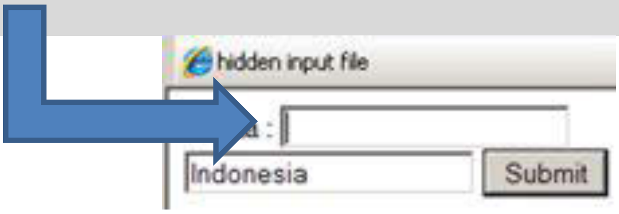
```
<input type="submit" value="Submit">
</form>
</body>
</html>
```



Gambar 1.7 hasil form input dengan hidden

Bila tag type="hidden" dihilangkan maka pada <input type="hidden"> maka menghasilkan tampilan sebagai berikut :

```
<form >
 Nama : <input type="text" name="fnama"><br>
        <input name="negara" value="Indonesia">
<input type="submit" value="Submit">
</form>
```



Gambar 1.8 hasil form input tanpa atribut hidden

5) Pembuatan form biodata dengan input text

Berikut ini adalah contoh dari sebuah halaman web yang berisi form yang sederhana :

```
<html>
<head>
<title> Form 1 </title>
</head>
<body>
<form name="form1" method="POST" action="inputdata.php">
```

```
     Nama : <input type="text" name="nama"><br><br>
     Kelas: <input type="text" name="kelas"
maxlength="2"><br><br>
     No   : <input type="text" name="nomor"
value="10"><br><br>
     Hobby: <input type="text" name="hobby"
size="8"><br><br>
     Password: <input type="password" name="password">
<br><br>
<input type="submit" name="submit"
value="Submit">   <input type="reset"
name="reset" value="Reset">
</form>
</body>
</html>
```

Apabila dibuka di dalam browser, maka tampilan dari susunan kode HTML di atas adalah sebagai berikut :



Gambar 1.9 pembuatan form input text sederhana

6) Pembuatan form dengan elemen fieldset

Berikut adalah contoh pembuatan form dengan menggunakan elemen fielsdet serta input form :

```
<!DOCTYPE html>
<html>
<head>
<title>Hooya</title>
</head>
<body>
<fieldset >
<legend>Proses Sign in</legend>
<td width="256" align="center" valign="top"><p
align="center"><font
color="#000000" size="+3">YAHOO!
<label for="textfield" ></label>
</font><font color="#000000" size="+3">
<label for="textfield" ></label>
</font><font size="+3">
<label for="textfield" ></label>
</font>
<label for="textfield" ></label>
</p>
<p align="center">
<input type="text" name="textfield" id="textfield"
value="ID
Yahoo" />
</p>
<p align="center">
<label for="textfield2"></label>
<input type="text" name="textfield2" id="textfield2"
value="Password" />
</p>
<p align="center">
<input type="submit" name="button" id="button"
value="Sign in"
/>
</p>
```

```
<p align="center" ><font color="#0000FF">Tidak bisa
mengakses
account
</font></p>
<p align="center" ><font color="#0000FF">Bantuan Sign
in</font></p><p
align="center">------------Atau ------------ </p>
<p align="center">
<input type="submit" name="button2" id="button2"
value="Buat
Account Baru"
/>
</p>
<p align="center">------------------------------</p>
<p align="center"> Masuk dengan Facebok atau Google</p>
<p> </p></td>
</fieldset>
</body>
</html>
```

Bila listing code diatas dijalankan akan menghasilkan tampilan sebagai berikut :



Gambar 1.10 pembuatan form input fieldset

c. Rangkuman.

Pada kegiatan belajar menyajikan komponen entri teks dapat disimpulkan menjadi beberapa point penting seperti berikut :

- Data bisa diinputan dalam bentuk textarea, input teks , pilihan radio button, check box dan lain sebagainya
- Komponen text area merupakan area tempat untuk menyimpan text atau tulisan baru
- Atribut – atribut pada textarea diantaranya, autofocus, cols, disabled, form, maxlength, name, placeholder, readonly, required, rows, wrap
- Komponen input text password bertanggung jawab untuk memasukan data password.
- Komponen input text merupakan komponen untuk memasukan data text ke server dalam bentuk textfield
- Pada format form input hidden ditandai dengan atribut form type = hidden

d. Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1. Buatlah form dengan menggunakan seperti berikut ini



2. Buatlah form dengan menggunakan elemen fieldset seperti contoh berikut ini



3. Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e. Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Jelaskan tentang 6 atribut baru pada komponen textarea html5 ?

2 Bagaimana format penulisan untuk pembuatan textarea multiline ?

3.Bagaimana format penulisan input password ?

4. Bagaimana format penulisan input text hidden ?

f. Lembar Jawaban Tes Formatif.

LJ- 01 : 6 atribut baru pada komponen textarea html5

....

....

....

....

....

....

....

....

....

....

LJ- 02 : format penulisan untuk pembuatan textarea multiline

....

....

....

....

....

....

....

....

....

LJ- 03 : format penulisan input password

....
....
....
....
....
....
....
....
....
....

LJ- 04 : format penulisan input text hidden

....
....
....
....
....
....
....
....
....
....

g. Lembar Kerja Siswa.

2. Kegiatan Belajar 2 : Menyajikan Komponen Entri Pilihan

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 2 diharapkan peserta didik dapat :

1) Memahami komponen entri pilihan
2) Menyajikan komponen entri pilihan ke dalam sebuah halaman Web

b. Uraian materi.

Form HTML digunakan untuk memilih pelbagai macam inputan yang disediakan. Data bisa diinputan dalam komponen entri pilihan, diantaranya input file, radio button, chexbox, select serta datalist.

1.) Komponen input file

Komponen input file berfungsi untuk memasukan data file yang dibutuhkan kedalam sistem web yang dibuat. Format tag HTML-nya adalah sebagai berikut :

```
<input type=file name=name accept=mime type list>
```

Name= “name” merupakan pemberian nama pada komponen input sebagai nama dari komponen. Karena pada saat sistem web dijalankan maka data yang diolah di dalam pemrograman adalah berdasarkan nilai (value) dari nama komponen input tersebut. Contoh bentuk tampilannya adalah sebagai berikut :

```
<!doctype html>
<html>
 <head>
<title>input file </title>
</head>
<body>
<font face="Courier New, Courier, mono">---- Tambahkan
File KRS --- </font><br>
<form >
<input type="file" name="pic" accept="image/*">
</form>
 </body>
```

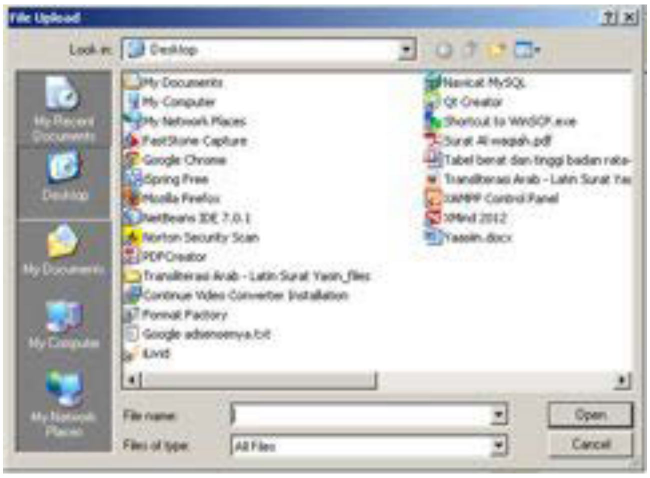
```
</html>
```

Listing kode diatas menghasilkan tampilannya sebagai berikut :



Gambar 2.1 hasil form input file

Bila dklik browse maka akan muncul kotak dialog untuk mencari file, seperti berikut :



Gambar 2.2 file upload dari form input

2.) Komponen radio button

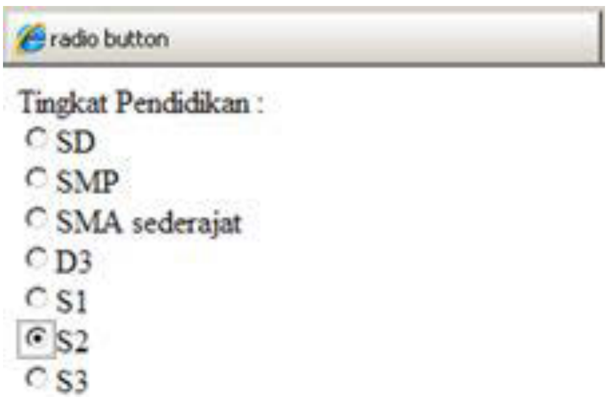
Pada form input radio button hanya ada satu pilihan yang terseleksi,tidak bisa digunakan untuk multiple answer. Format umum tag HTML dari Form Input Radio Button adalah sebagai berikut :

```
<input type=radio name=name value=value>
<input type=radio name=name value=value checked>
```

Name= “name” merupakan pemberian nama pada komponen input sebagai nama dari komponen. Karena pada saat sistem web dijalankan maka data yang diolah di dalam pemrograman adalah berdasarkan nilai (value) dari nama komponen input tersebut. Berikut adalah penggalan listing program penulisan form input radio button

```
<!doctype html>
<html>
<head>
<title>radio button </title>
</head>
<body>
Tingkat Pendidikan :
<form>
<input type="radio" name="pendidikan" value="SD">SD<br>
<input type="radio" name="pendidikan" value="SMP">SMP<br>
<input type="radio" name="pendidikan" value="SMA">SMA
sederajat<br>
<input type="radio" name="pendidikan" value="D3">D3<br>
<input type="radio" name="pendidikan" value="S1">S1<br>
<input type="radio" name="pendidikan" value="S2">S2<br>
<input type="radio" name="pendidikan" value="S3">S3<br>
</form>
</body>
 </html>
```

Yang perlu diperhatikan pada penggunaan input radio adalah bahwa atribut nama (name) dari komponen dari input tersebut harus sama. Contoh bentuk tampilannya adalah sebagai berikut :



Gambar 2.3 tampilan radio button

3) Komponen chexbox

Komponen chexbox memiliki fungsi yang hampir sama dengan radio button, yaitu untuk menentukan pilihan. Namun pada chexbox dapat digunakan untuk memilih lebih dari satu jawaban (multi answer).

Atau bisa dikatakan bahwa komponen input checkbox digunakan sebgai masukan untuk data atau nilai pilihan dengan alternatif jawaban lebih dari 1 opsi pilihan. Tampilan dari komponen input ini berupa kotak yang dapat diberi tanda centang/contreng. Jenis input ini biasa digunakan untuk memasukkan data pilihan yang pilihannya bisa lebih dari satu, seperti hobby, makanan, dan lain-lain. Format tag HTML-nya adalah sebagai berikut :

```
<input type=checkbox name=name value=value>

<input type=checkbox name=name value=value checked>
```

Setiap komponen input harus diberikan nama (name) yang berbeda karena data yang diolah di dalam pemrograman adalah berdasarkan nilai (value) dari nama komponen input tersebut. Nilai dari komponen checkbox berupa nilai boolean, yakni true apabila dicentang dan false apabila input pilihan tersebut tidak dicentang.

Berikut ini adalah contoh penerapan dari format form input check box

```
<!doctype html>

<html>

<head>

<title>FORM INPUT</title>

</head>

---- KETERANGAN ----

<form>

<input type="checkbox" name="vehicle" value=" Sakit "
>Sakit<br>

<input type="checkbox" name="vehicle"
```

```
value="Ijin">Ijin<br>
<input type="checkbox" name="vehicle" value="Tanpa">Tanpa
Keterangan<br>
</form>
</body>
</html>
```

Listing kode diatas menghasilkan tampilannya sebagai berikut :



Gambar 2.4 hasil form input checkbox

4) Komponen Input Image

Komponen input image bertujuan untuk memasukan file gambar yang dibutuhkan oleh sistem web, misalnya saja file gambar foto.jpg. Format tag HTML dengan atribut image adalah sebagai berikut :

```
<input type=image name=name src="url">
<input type=image name=name src="url" align="alignment">
```

Name= “name” merupakan pemberian nama pada komponen input sebagai nama dari komponen. Karena pada saat sistem web dijalankan maka data yang diolah di dalam pemrograman adalah berdasarkan nilai (value) dari nama komponen input tersebut. Sedangkan type=image merupakan tipe file yang dapat dimasukan dalam sistem. Berikut adalah salah satu contoh form Input Image dengan atribut image

```
<!doctype html>
<html>
```

```
<body>

<form >

 Nama depan   : <input type="text" name="fname"><br>

 Nama belakang: <input type="text" name="lname"><br>

<input type="image" src="img_submit.gif" alt="Submit"

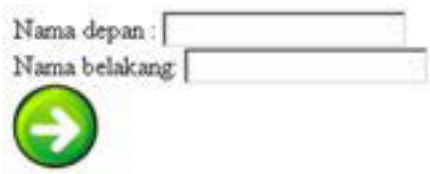
width="50" height="50">

</form>

</body>

</html>
```

Bila listing program diatas di jalankan di browser (http://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_input_type_image) akan menghasilkan tampilan sebagai berikut :



Gambar 2.5 form input image

5) Komponen select

Komponen input dengan tipe 'select' adalah komponen input dengan banyak pilihan nilai dan hanya memungkinkan untuk memilih 1 alternatif opsi pilihan. Istilah lain dari bentuk komponen ini adalah combo box atau list box. Contoh penggunaan komponen ini adalah pemilihan data agama, jurusan, pekerjaan, dan sebagainya. Format umum Penulisan komponen select adalah sebagai berikut :

```
 <select name=”name”>
  <option value="nilai1"> nilai1 </option>
  <option value=" nilai2"> nilai2 </option>
  <option value=" nilai3"> nilai3 </option>
```

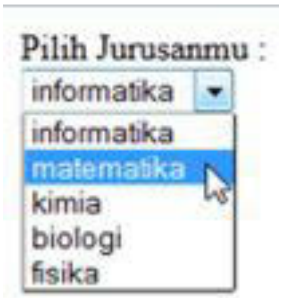
```
 <option value=" nilai4 "> nilai4</option>
</select>
```

Name= “name” merupakan pemberian nama pada komponen input select sebagai nama dari komponen. Karena pada saat sistem web dijalankan maka data yang diolah di dalam pemrograman adalah berdasarkan nilai (value) dari nama komponen input tersebut. Sedangkan value=”nilai” merupakan nilai dari setiap pilihan yang diberikan.

Berikut ini adalah contoh penggunaan dari komponen input bertipe ‘select’ :

```
<html>
<head>
<title>Form</title>
</head>
<body>
Pilih Jurusanmu :
<form>
<select name="jurusan">
<option value="informatika">informatika</option>
<option value="matematika">matematika</option>
<option value="kimia">kimia</option>
<option value="biologi">biologi</option>
<option value="fisika">fisika</option>
</select>
</form>
</body>
</html>
```

Hasil dari halaman form di atas adalah sebagai berikut :



Gambar 2.5 komponen form select

6) Komponen datalist

Komponen datalist merupakan elemen form baru pada HTML5. Datalist digunakan untuk membuat daftar list. Komponen datalist memberikan dukungan autocomplete dasar untuk pilihan pada field input. Komponen datalist merupakan pengabungan antara list dan field input dengan atribut autocomplete. Saat membuat pilihan pada datalist, bila terdapat huruf/item yang sama dengan data pada data list maka akan muncul item yang cocok (fungsi autocomplete bekerja).

Format umum Penulisan komponen datalist adalah sebagai berikut :

```
<input list="list">
<datalist id="id">
 <option value="nilai1">
 <option value=" nilai2">
 <option value=" nilai3">
</datalist>
```

Berikut ini adalah contoh penggunaan dari komponen input datalist

```
<!DOCTYPE html>

<html>

<body>

<form action="demo_form.asp" method="get">

<input list="browsers" name="browser">

<datalist id="browsers">

<option value="Internet Explorer">
```

```
<option value="Firefox">

<option value="Chrome">

<option value="Opera">

<option value="Safari">

</datalist>

<input type="submit">

</form>

</body>

</html>
```

Bila listing kode diatas dijalankan maka akan menghasilkan tampilan seperti dibawah ini. Pada saat textfield diketikan huruf “C” maka akan muncul pilihan “chrome” dimana kata “chrome” menggandung unsur huruf “C” hal ini menandakan fungsi autocompletenya berjalan.



Gambar 2.7 komponen form datalist

7) Contoh Pembuatan Form dengan elemen entri pilihan

Berikut adalah listing kode untuk membuat form menggunakan elemen entri pilihan, textarea

```
<!DOCTYPE html >

<html>

<head>

<title>form</title>

</head>

<body>
```

```
<table border="1">
<tr><td><form id="form1" name="form1" method="post"
action="">
<table width="354"  align="center">
<tr>
<td height="24" colspan="3"
align="center"><B>PENDAFTARAN</td>
</tr>
<tr>
<td>Nama Pengguna</td>
<td align="center">:</td>
<td><label for="textfield"></label>
<input type="text" name="textfield" id="textfield"
/></td>
</tr>
<tr>
<td>Sandi</td>
<td align="center">:</td>
<td><label for="textfield2"></label>
<input type="text" name="textfield2" id="textfield2"
/></td>
</tr>
<tr>
<td>Jenis Kelamin</td>
<td align="center">:</td>
<td><input type="checkbox" name="checkbox" id="checkbox"
```

```
/>
<label for="checkbox">Pria
<input type="checkbox" name="checkbox2" id="checkbox2" />
      Wanita</label></td>
</tr>
<tr>
<td>Tanggal lahir</td>
<td align="center">:</td>
<td><label for="select"></label>
<select name="select" id="select">
<option>1</option>
<option>2</option>
<option>3</option>
<option>4</option>
<option>5</option>
<option>6</option>
<option>7</option>
<option>8</option>
<option>9</option>
<option>10</option>
</select>
<label for="select2"></label>
<select name="select2" id="select2">
<option>Januari</option>
<option>Februari</option>
<option>Maret</option>
```

```
<option>April</option>
<option>Mei</option>
<option>Juni</option>
<option>Juli</option>
<option>Agustus</option>
<option>September</option>
<option>Oktober</option>
<option>November</option>
<option>Desember</option>
</select>
<label for="select3"></label>
<select name="select3" id="select3">
<option>1991</option>
<option>1992</option>
<option>1993</option>
<option>1994</option>
<option>1995</option>
<option>1996</option>
<option>1997</option>
<option>1998</option>
<option>1999</option>
</select></td>
</tr>
<tr>
<td colspan="2"> </td>
<td><input type="submit" name="button" id="button"
```

```
value="Simpan" />
<input type="reset" name="button2" id="button2"
value="Batal" /></td>
</tr>
</table>
</td></tr></table>
</form>
</body>
</html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan form dilengkapi elemen entri pilihan seperti gambar dibawah ini.



Gambar 2.8 elemen form entri pilihan

8) Pembuatan form dengan elemen entri pilihan dan elemen fieldset

Berikut adalah listing kode untuk membuat form menggunakan elemen entri pilihan, textarea dan fieldset.

```
<html>
<head>
<title>pengaturan pada form</title>
</head>
```

```
<body>
<form>
<form action="test.html">
<fieldset>
<legend>tentang anda </legend>
<table>
<tr>
<td width="50">
<label for="first">nama lengkap </label>
</td>
<td><input type="text" name="nama_lengkap" id="first"
size="20" maxlength="50" /><br />
</td>
</tr>
<tr><td><label for="last">alamat</label>
</td>
<td><input type="text" name="alamat" id="last" size="20"
maxlength="50" /><br />
</td>
</tr>
<tr><td><label for="desc">pesan dan kesan</label>
<td><textarea rows="5" cols="20" id="desc">tuliskan pesan
anda disini</textarea><br />
</td>
</tr>
</table>
```

```
<fieldset class="hoby">
      <legend>hoby anda?</legend>
      <input type="radio" id="berkebun" value="berkebun"
name="hoby" /><label for="cberkebun"> berkebun</label><br
/>
      <input type="radio" id="memasak" value="memasak"
name="hoby" /><label for="memasak"> memasak</label><br />
      <input type="radio" id="renang" value="renang"
name="hoby" /><label for="renang"> renang</label><br />
</fieldset>
<input type="submit" value="submit" id="submit" />
</fieldset>
</form>
</body>
</html>
```

Bila listing kdoe diatas jalankan akan menghasilkan tampilan sebagai berikut :



Gambar 2.9 elemen form entri dan fieldset

c. Rangkuman.

Pada kegiatan belajar menyajikan komponen entri pilihandapat disimpulkan menjadi beberapa point penting seperti berikut :

- Data bisa diinputan dalam komponen entri pilihan, diantaranya input file, radio button, chexbox, select serta datalist.
- Komponen input file berfungsi untuk memasukan data file yang dibutuhkan kedalam sistem web yang dibuat
- Format tag HTML input file adalah sebagai berikut : <input type=file name=name accept=mime type list>
- Pada form input radio button hanya ada satu pilihan yang terseleksi
- Format tag HTML input radio button adalah sebagai berikut <input type=radio name=name value=value>
- komponen input checkbox digunakan sebgai masukan untuk data atau nilai pilihan dengan alternatif jawaban lebih dari 1 opsi pilihan.
- Format tag HTML input checkbox adalah sebagai berikut <input type=checkbox name=name value=value>
- Komponen input image bertujuan untuk memasukan file gambar yang dibutuhkan oleh sistem web, misalnya saja file gambar foto.jpg.

- Format tag HTML dengan atribut image adalah sebagai berikut : <input type=image name=name src="url">
- Komponen input dengan tipe ‘select’ adalah komponen input dengan banyak pilihan nilai dan hanya memungkinkan untuk memilih 1 alternatif opsi pilihan
- Komponen datalist memberikan dukungan autocomplete dasar untuk pilihan pada field input
- Name= “name” merupakan pemberian nama pada komponen input select sebagai nama dari komponen. Karena pada saat sistem web dijalankan maka data yang diolah di dalam pemrograman adalah berdasarkan nilai (value) dari nama komponen input tersebut.

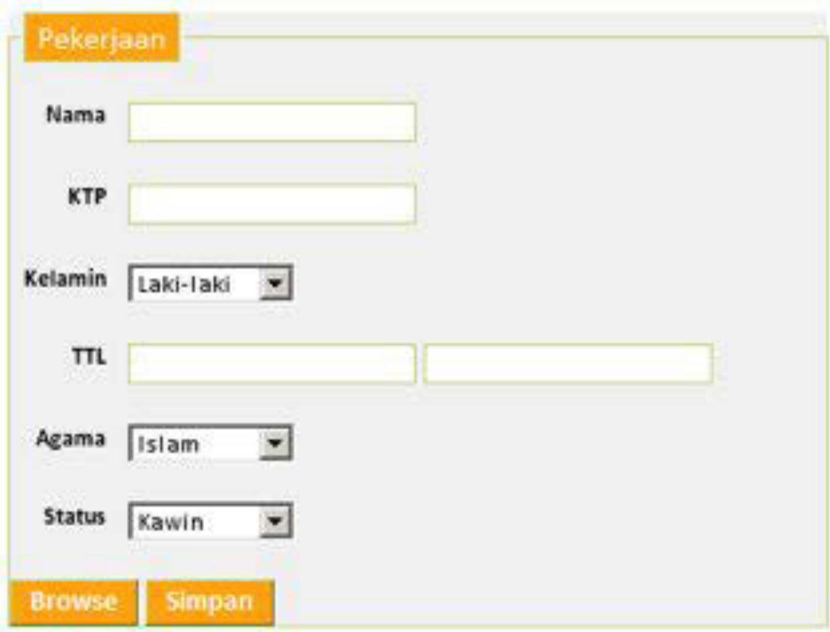
d. Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1. Buatlah form dengan menggunakan seperti berikut ini



2. Buatlah form dengan menggunakan elemen fieldset seperti contoh berikut ini



3. Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e. Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Jelaskan fungsi dan format penulisan komponen input file
2. Jelaskan fungsi dan format penulisan komponen radio button
3. Jelaskan fungsi dan format penulisan komponen chexbox
4. Jelaskan fungsi dan format penulisan komponen datalist

f. Lembar Jawaban Tes Formatif.

LJ- 01 : komponen input file

....

....

....

....

....

....

....

....

....

....

LJ- 02 : komponen radio button

....

....

....

....

....

....

....

....

....

LJ- 03 : komponen chexbox

....

....

....

....

....

....

....

....

....

....

..**LJ- 04 :** komponen datalist

....

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

3. Kegiatan Belajar 3 : Memahami Anatomi dan Cara Kerja Cascading Style Sheet

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 3 ini siswa diharapkan dapat :

1) Mengetahui penggunaan cascading style sheet
2) Memahami anatomi cascading style sheet
3) Mengetahui cara kerja cascading style sheet
4) Memahami cara kerja cascading style sheet

b. Uraian materi.

1) Definisi dan fungsi cascading style sheet

Cascading Style Sheet atau lebih sering disebut dengan istilah CSS merupakan salah satu dokumen website yang bertujuan untuk mengatur gaya (style) tampilan website.

CSS bukanlah sebuah bahasa pemrograman, melainkan sebuah aturan untuk mengendalikan beberapa komponen pada web sehingga akan lebih terstruktur, seragam dan mudah dalam pengaturan saat pendesain sebuah halaman web.
CSS merupakan sekitar pembuatan dan pengaturan style font, warna, jarak, bentuk, dan lain-lain yang tidak dapat dilakukan dengan elemen - elemen HTML. Dapat dikatakan bahwa CSS merupakan pengembangan dari HTML dalam hal format dokumen web. Adanya CSS memudahkan kita untuk mengatur dan memilih sebuah website dan tampilannya karena CSS memisahkan antara bagian presentasi dan isi dari web yang dibuat.
Selain itu dengan penggunaan dari CSS dalam pembuatan web akan memberikan beberapa manfaat seperti berikut ini :

- Kode HTML menjadi lebih sederhana dan lebih mudah diatur.
- Ukuran file menjadi lebih kecil sehingga load file lebih cepat.
- Mudah untuk mengubah tampilan, hanya dengan mengubah file CSS saja.
- Dapat berkolaborasi dengan JavaScript dan merupakan pasangan setia HTML.
- Dapat digunakan dalam hampir semua jenis web browser.

CSS merupakan sebuah teknologi internet yang di rekomendasikan oleh World Wide Web Consortium atau W3C pada tahun 1996. Setelah CSS di standarisasikan, internet explorer dan Netscape merilis browser terbaru yang mendukung standar CSS. Terdapat tiga versi CSS, CSS1, CSS2, dan CSS3. Yaitu :

Versi CSS	Fitur CSS
CSS 1	Fokus mengatur pemformatan dokumen HTML, seperti - Font (Jenis ketebalan). -Warna, teks, background dan elemen lainnya. - Text attributes, misalnya spasi antar baris, kata dan huruf. - Posisi teks, gambar, table dan elemen lainnya. - Margin, border dan padiing.
CSS 2	- Mengatur format dokumen untuk kebutuhan di cetak dengan printer - Posisi konten - Downloadable - Font huruf - Table layout - Media tipe yang
CSS 3	-Sangat mendukung tampilan desain website - Animasi warna bahkan sampai animasi 3D - Menunjang kompabilitas website dengan smartphone - CSS math - CSS obyek model - Mendukung fungsi multimedia pada website - Beberapa efek teks, seperti teks berbayang, kolom koran, dan "word-wrap" -Jenis huruf eksternal, sehingga dapat menggunakan huruf yang tidak termasuk "web-safe fonts". -Beberapa efek pada kotak, seperti

	kotak yang ukurannya dapat diubah-ubah, transformasi 2 dimensi dan 2 dimensi, sudut tumpul dan shadow

CSS 3 merupakan versi CSS terbaru yang masih dikembangkan oleh W3C. Namun beberapa web browser sudah mendukung CSS 3. CSS 2 didukung seutuhnya oleh CSS 3 dan tidak ada perubahan, hanya ada beberapa penambahan. Ketika sebuah web yang semula menggunakan CSS 2 bermigrasi CSS 3, tidak perlu mengubah apapun.

Dengan adanya CSS mempermudah pengaturan dan pemeliharaan sebuah website dan tampilannya karena CSS memisahkan antara bagian pengatur gaya dan isi dari web.

2) Anatomi dari cascading style sheet

Bagian – bagian utama dari sebuah CSS itu terdiri dari tiga bagian yaitu: selector, property, value. Jika dalam HTML dikenal ada tag-tag yang menyusun sebuah dokumen HTML maka dalam CSS juga dikenal dengan sebutan selector.

Bagian CSS	Keterangan
Selector	- nama-nama yang diberikan untuk style-style yang berbeda, baik itu style internal maupun eksternal - bagian elemen HTML yang akan ditunjuk untuk mengatur style - dapat berupa class dan ID
Property	- aturan dalam CSS untuk mengubah selector yang dipilih - property mempunyai nilai yang disebut dengan value

	-	properties di dalam tanda { }
value.	-	Merupakan nilai dari property CSS

CSS memiliki aturan tersendiri dalam penulisan sintaks. Penulisan sintaks tersebut terbagi ke dalam tiga bagian yaitu: selector, property, dan value. Format penulisan bagian-bagian dari CSS dapat dituliskan sebagai berikut :

```
Selector{ Property : value; }
```

Berikut ini sedikit contoh penggunaan selektor, property, dan value untuk mengatur style pada sebuah form

```
form{ margin-left:0; }
```

Dari contoh penggunaan selector, property, dan value di atas maka tiap bagian akan di jelaskan sebagai berikut :

- form : selector yang akan di atur stylenya
- margin-left : property yang digunakan untuk jarak fieldset dari batas kiri layout
- 0 : nilai dari property margin-left

Dengan adanya CSS memungkinkan untuk menampilkan halaman yang sama dengan format yang berbeda.

3) Cara Kerja cascading style sheet

Cara kerja CSS dimulai saat deklarasi style yang diinginkan dengan menulis style (selektor, id dan class), maka secara otomatis style tersebut akan bekerja pada dokumen HTML. Dengan mengatur selektor, id, dan class, untuk kemudian id dan class tersebut kita sesuaikan dan terapkan pada kode HTML . Dan secara otomatis pula CSS itu akan bekerja pada file HTML.

Pendefinisian style bisa dilakukan pada tag <style>. Di dalam pasangan tag tersebut, pendefinisian style dilakukan dengan bentuk `Selector { … }` .

Dalam hal ini, selector bisa berupa selector HTML, selector kelas, atau selector ID.

Selector HTML mendefinisikan style bagi suatu tag HTML. Contoh di depan menggunakan selector HTML beruma img. Bentuk umumnya :

Contoh :

```
Nama-elemen { … }h:
```

Contoh :

```
Img { Margin-topi: 10px; Float: left; }
```

Selektor class mendefinisikan kelas yang bisa berlaku untuk sebarang tag HTML. Bentuknya:

Contoh :

```
.nama-class { … }
```

Contoh :

```
.kotak { Border: solid; }
```

Selector ID mendefinisikan style bagi elemen yang memiliki ID sesuai yang tercantum dalam selector. Bentuknya:

```
#id { … }
```

Contoh :

```
#inggris {Font-weight:bold;Font-
size:1.2em;}
```

Berikut contoh yang menggunakan selector class dan selector ID dapat dilihat di bawah ini.

```
<html>
<head>
<tittle>Contoh selektor kelas dan ID</tittle>
<style type=”text/css”>
     .kotak { Border: solid; Padding: 5px; }
     #jawa { Background-color: #ccff66; /* latarbelakang
*/}
     #inggris { Font-weight: bold; Font-size: 1.2em; }
</style>
</head>
```

```
<body>
<div id=”Indonesia” class=”kotak”>Selamat pagi !</div>
<div id=”jawa”>Sugeng enjing !</div>
<div id=”inggris” class=”kotak”>Good morning !</div>
</body>
</html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut :



Gambar 3.1 hasil penggunaan selektor class dan id

Berdasarkan letak tempat stylenya, penerapan CSS pada sebuah halaman web dapat dilakukan dengan beberapa alternatif di antaranya adalah :

A. Inline Style Sheet

CSS didefinisikan langsung pada tag HTML yang bersangkutan. Cara penulisannya cukup dengan menambahkan atribut style="..." dalam tag HTML tersebut. Style hanya akan berlaku pada tag yang bersangkutan, dan tidak akan mempengaruhi tag HTML yang lain. Contoh penulisan CSS secara inline style sheet, CSS berikut mengatur style di elemen <p></p> atau paragraf.

```
<p align="center" style="color:#303; font-size:24px">SMK
</p>
```

Saat penulisan CSS diatas, listing kode CSS langsung dipasang di tag HTML, namun hal tidak direkomendasikan karena akan menyulitkan pengaturan style jika terjadi perubahan tampilan website dikemudian hari.

B. Embedded Style Sheet

CSS didefinisikan terlebih dahulu dalam tag `<style> ... </style>` di atas tag `<body>`, lebih tepatnya didalam tag `<head>`

```
</style>
</head><html>
<head>
<style>
input.btn
    {color: #fff;
     background: #ffa20f;
     border: 2px outset #BBD16D;
     font-family: "calibri", Times, serif;
     font-size: 14px;
     font-weight: bold;
     width : auto;}
<body>
<table>
<tr>
<td><form >
<p><label for="nama">Username</label>
<input type="text" id="nama" />
</p>
<p>
<label for="KTP">Password</label>
<input type="text" id="KTP" />
</p>
<p>
<input type="submit" value="Login" class="btn"
onClick=parent.location="Home_Iframe.htm"  />
</p>
</form></td>
</tr>
</table>
```

```
</body>
</html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan sebagai berikut ini :



Gambar 3.2 penerapan embedded style sheet

Pada listing kode diatas, penerapan style diberikan pada komponen input button hal ini ditandai dengan pembuatan CSS dengan nama “input.btn” didalam tag <style></style>

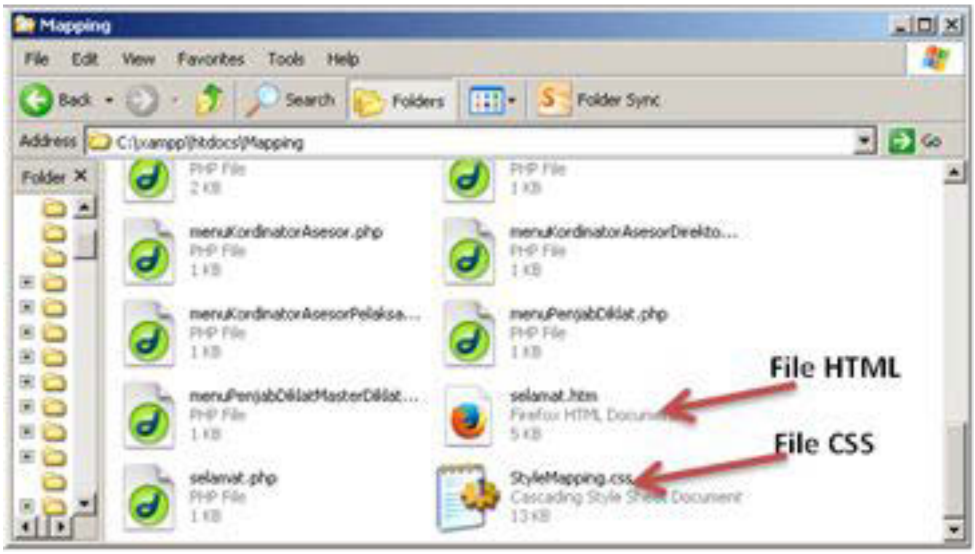
```
<style>
input.btn
    { ……. }
```

Di bagian body terdapat penerapan CSS dengan cara memanggil nama class nya

C. External Style Sheet

```
<input type="submit" value="Login" class="btn"
onClick=parent.location="Home_Iframe.htm" />
```

CSS didefinisikan secara terpisah pada file yang berbeda. Dan selanjutnya file atau halaman web yang ingin menerapkan style pada file CSS tersebut tinggal memanggil file CSS tersebut. Cara ini umumnya digunakan pada satu aplikasi berbasis web sehingga tampilan antar halaman akan terlihat konsisten.



Gambar 3.3 file .HTML dan .CSS

File yang berisi style CSS disimpan dalam format *.css. Dari gambar diatas dapat dilihat file CSS ditempatkan dalam satu folder yang sama dengan file HTML. Contoh penulisan file.css

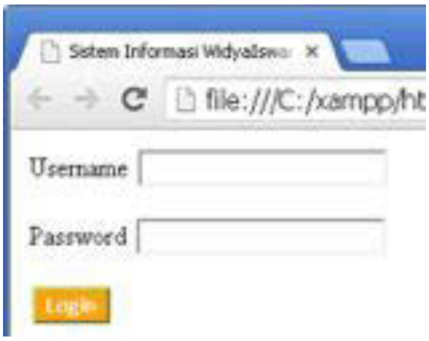
```
input.btn
    {color: #fff;
    background: #ffa20f;
    border: 2px outset #BBD16D;
    font-family: "calibri", Times, serif;
    font-size: 14px;
    font-weight: bold;
    width : auto;}
```

Kemudian listing kode diatas disimpan dengan nama style.css dan untuk selanjutkan akan dipanggil pada file html seperti pada contoh berikut :

```
<html>
<head>
<title>Sistem Informasi WidyaIswara</title>
<style type="text/css">
</style>
```

```
<link href="style.css" rel="stylesheet" type="text/css">
</head>
<body>
<table>
<tr>
<td><form >
<p><label for="nama">Username</label>
<input type="text" id="nama" />
</p>
<p>
<label for="KTP">Password</label>
<input type="text" id="KTP" />
</p>
<p>
<input type="submit" value="Login" class="btn"
```

Bila listing kode diatas dijalankan maka akan menghasilkan tampilan seperti berikut :



Gambar 3.4 penerapan sxternal style sheet

Pada bagian head terdapat perintah pemanggilan file “style.css” seperti pada listing kode berikut :

```
<style type="text/css">
</style>
<link href="style.css" rel="stylesheet" type="text/css">
```

c. Rangkuman.

Pada kegiatan belajar memahami anatomi dan cara kerja cascading style sheet dapat disimpulkan menjadi beberapa poin penting seperti berikut :

- CSS merupakan salah satu dokumen website yang bertujuan untuk mengatur gaya (style) tampilan website.
- Beberapa manfaat dari penggunaan CSS
  - Kode HTML menjadi lebih sederhana dan lebih mudah diatur.
  - Ukuran file menjadi lebih kecil sehingga load file lebih cepat.
  - Mudah untuk mengubah tampilan, hanya dengan mengubah file CSS
  - Dapat digunakan dalam hampir semua jenis web browser.
- CSS 3 merupakan versi CSS terbaru yang masih dikembangkan oleh W3C dan CSS 2 didukung seutuhnya oleh CSS 3 dan tidak ada perubahan
- Anatomi CSS terbagi ke dalam tiga bagian yaitu: selector, property, dan value.

```
Selector{ Property : value; }
```

- Cara kerja CSS dimulai saat deklarasi style yang diinginkan dengan
- menulis style (selektor, id dan class)
- Format penulisan CSS dengan Selector HTML adalah Nama-elemen
- Format penulisan CSS dengan Selector class adalah .nama-class { … }
- Format penulisan CSS dengan Selector id adalah #id { … }
- Penerapan CSS pada sebuah halaman web dapat dilakukan dengan
- beberapa alternatif di antaranya adalah: inline style sheet, embedded style sheet, external style sheet

d. Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1. Jelaskan tentang anatomi cascading style sheet
2. Jelaskan tentang macam selector yang dapat digunakan pada penerapan cascading style sheet
3. Jelaskan cara kerja CSS pada penulisan style dengan cara inline style sheet, embedded style sheet, external style sheet

Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e. Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Tuliskan anatomi cascading style sheet pada sebuah dokumen html
2. Sebutkan macam selector cascading style sheet
3. Jelaskan keuntungan penerapan cascading style sheet
4. Berikan contoh penerapan cascading style sheet yang ditulis secara inline style sheet pada sebuah elemen html.

f. Lembar Jawaban Tes Formatif.

LJ- 01 : anatomi cascading style sheet

....

....

....

....

....

....

....

....

....

....

LJ- 02 : macam selector cascading style sheet

....

....

....

....

....

....

....

....

....

LJ- 03 : keuntungan penerapan cascading style sheet

....

....

....

....

....

....

....

....

....

....

LJ- 04 : dengan inline style sheet

....

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

4. Kegiatan Belajar 4 : Menyajikan Cascading Style Sheet untuk Memformat Tampilan Teks

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 4 diharapkan peserta didik dapat :

1) Memahami style-style dasar CSS untuk memformat tampilan teks
2) Menerapkan beberapa style untuk memformat tampilan teks.
3) Menyajikan CSS untuk memformat tampilan teks pada sebuah halaman web

b. Uraian materi.

Agar lebih menarik, mudah dalam hal pengaturans serta editing, teks pada sebuah halaman web dapat diberikan sytle dalam kemasan CSS. Pengaturan style tersebut meliputi banyak properti, diantaranya adalah pewarnaan, pemilihan font, spasi dan lain sebagainya. Untuk lebih detailnya dapat dilihat pada tabel berikut ini.

Properti	Keterangan
Color	Mengatur warna dari teks
Direction	Menentukan arah tulisan/teks
letter-spacing	Menambah ataupun mengurangi spasi antar karakter dalam teks
line-height	Mengatur tinggi baris teks
text-align	Menentukan batas teks secara horisontal
text-decoration	Menentukan dekorasi/hiasan pada teks
text-indent	Menentukan jarak inden dari baris pertama dalam teks-blok
text-shadow	Menentukan efek bayangan pada teks
text-transform	Mengatur huruf besar-huruf kecil dari teks

unicode-bidi	Untuk mengeset unicode
vertical-align	Menentukan batas teks secara horisontal
white-space	Menentukan penulisan white-space pada elemen
word-spacing	Menambah ataupun mengurangi spasi antar kata dalam teks

1) Pengaturan warna pada teks/color

CSS dapat terapkan untuk mengatur warna dari teks. Pengaturan warna teks dengan CSS dapat menggunakan nilai warna sebagai berikut :

- nilai HEX → misalnya, "#ff0000"
- nilai RGB→ misalnya, "rgb(255,0,0)"
- nama dari warna tersebut→misalnya "red"

Format penulisan untuk pemberian warna pada text menggunakan CSS adalah sebagai berikut :

```
selector {color:nilai warna}
```

Berikut adalah contoh penulisan warna pada teks yang dituliskan secara embeded

```
<!DOCTYPE html>
<html>
<head>

<style>
 body {color:red;}
h1 {color:#00ff00;}
p.ex {color:rgb(0,0,255);}
</style>

</head>
<body>
<h1>ini adalah heading 1</h1>
```

Blok CSS

```
 <p>ini adalah paragraph, ditulis dengan warna merah.
 Default text-color didefinisakan pada body
selector.</p>
<p class="ex">ini adalah paragraph with class="ex".
dan warna teks biru.</p>
</body>
 </html>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut, secara otomatis ketika terdapat teks pada tag <body., <h1> dan <p> terkena style pewarnaan.



Gambar 4.1 penerapan CSS warna teks

2) Pengaturan spasi antar karakterf/letter-spacing

Salah satu style dari CSS adalah letter-spacing yang dapat digunakan untuk memberikan jarak (spasi) antar karakter atau huruf. Format penulisan CSS untuk pengaturan jarak (spasi) antar karakter atau huruf adalah sebagai berikut :

```
selector { letter-spacing:nilai spasi;}
```

Besarnya nilai spasi dapat ditentukan dalam satuan cm maupun px. Berikut ini adalah contoh penerapan dari style tersebut :

```
<HTML>
 <HEAD>
 <TITLE>Format Text </TITLE>
 <STYLE ="text/css">
```

```
p {letter-spacing: 0.5cm}
h4 {letter-spacing: -2px}
</STYLE>
</HEAD>
<BODY>
<p>Pengaturan Spasi Pada Paragraph</p>
<h4> Header 4</h4>
</BODY>
</HTML>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut,



Gambar 4.2 penerapan CSS untuk pengaturan spasi pada paragraf teks

3) Pengaturan bentuk-bentuk teks/decoration

Pengaturan bentuk teks menggunakan properti text-decoration bertujuan untuk mengatur atau menghapus dekorasi dari teks. Format penulisan CSS untuk mengaturan dekorasi teks adalah sebagai berikut :

```
selector {text-decoration : nilai text-decoration }
```

Nilai dari text-decoration dapat berupa overline, line-through, underline dan none yang berarti tanpa dekorasi . Berikut ini adalah contoh penerapan dari style tersebut :

```
<HTML>
<HEAD>
<TITLE>Format Text </TITLE>
    <STYLE ="text/css">
    em {text-decoration : overline}
    h2 {text-decoration: line-through}
    h3 {text-decoration: underline}
    a {text-decoration: none}
</STYLE>
```

```
</HEAD>
<BODY>
    <em>Bentuk Overline</em>
    <h2>Header 2, Bentuk Line-through</h2>
    <h3>Header 3,Bentuk Underline</h3>
<p><a href="http://www.bem.akakom.ac.id">
Penggunaan Dalam Link,Nilai NONE</a></p>
</BODY>
</HTML>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut,



Gambar 4.3 penerapan CSS untuk pengaturan dekorasi teks

4) Pengaturan spasi antar kata/word-spacing

Salah satu style dari CSS adalah word-spacing yang dapat digunakan untuk memberikan jarak (spasi) antar kata. Format penulisan CSS untuk pengaturan jarak (spasi) antar karakter atau huruf adalah sebagai berikut :

```
selector { word-spacing:nilai spasi;}
```

Besarnya nilai spasi dapat ditentukan dalam satuan cm maupun px. Berikut ini adalah contoh penerapan dari style tersebut :

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut,

```
<HTML>

<HEAD>

<TITLE>Format Text </TITLE>
```

```
     <STYLE ="text/css">
     p {word-spacing: 1cm}
     h2 {word-spacing: 5px}
</STYLE>
</HEAD>
<BODY>
     <p>Pengaturan Spasi per Kata sebesar 1 cm</p>
     <h2> Peregangan Header 2 sebesar 5 px</h2>
</BODY>
</HTML>
```



Gambar 4.4 penerapan CSS untuk pemberian efek bayangan

5) Pengaturan jarak indentasi paragraph/text-indent

Text-indent menentukan jarak inden dari baris pertama dalam teks-blok, sehingga teks pada baris pertama terlihat lebih menjorok kedalam dibanding teks pada baris yang lain. Format penulisan untuk pengaturan jarak indentasi text adalah sebagai berikut :

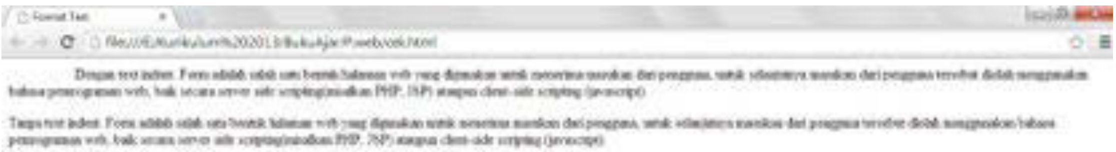
```
selector { text-indent:nilai indent}
```

Besarnya nilai spasi dapat ditentukan dalam satuan cm maupun px. Berikut ini adalah contoh penerapan dari style tersebut :

```
<HTML>
```

```
<HEAD>
<TITLE>Format Text </TITLE>
<STYLE ="text/css">
    p {text-indent: 2cm }
</STYLE>
</HEAD>
<BODY>
<p>
Dengan text indent.
Form  adalah  salah  satu  bentuk  halaman  web  yang  digunakan
untuk menerima masukan dari pengguna,
untuk selanjutnya masukan dari pengguna tersebut diolah
menggunakan bahasa pemrograman web,
baik secara server side scripting(misalkan PHP, JSP)
ataupun client-side scripting (javascript).
</p>
Tanpa text indent.
Form adalah salah satu bentuk halaman web yang digunakan
untuk menerima masukan dari pengguna,untuk selanjutnya
masukan dari pengguna tersebut diolah menggunakan bahasa
pemrograman web,baik secara server side scripting(misalkan
PHP, JSP)  ataupun client-side
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut :



Gambar 4.5 penerapan CSS untuk pengaturan text indent

6) Pengubahan huruf besar dan huruf kecil dari sebuah teks/text-transform

Teks-transform properti digunakan untuk menentukan huruf besar dan kecil dalam teks. Format penulisan CSS untuk pengaturan text indent adalah sebagai berikut :

```
selector { text-transform:nilai text transform}
```

Nilai text transform dapat diisi dengan uppercase,lowercase, capital. Pada listing kode berikut terlihat CSS dituliskan dalam bentuk CLASS p.besar, p.kecil dan p.kapital. Hal ini bertujuan untuk memberikan pengaturan yang berbeda pada paragrafnya.

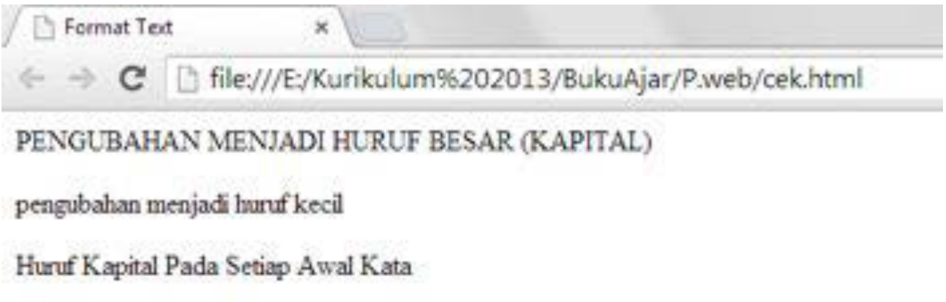
```
<STYLE ="text/css">
    p.besar {text-transform: uppercase}
    p.kecil {text-transform: lowercase}
    p.kapital {text-transform: capitalize}
</STYLE>
```

Berikut ini adalah contoh penerapan dari style tersebut :

```
<HTML>
<HEAD>
<TITLE>Format Text </TITLE>
<STYLE ="text/css">
    p.besar {text-transform: uppercase}
    p.kecil {text-transform: lowercase}
    p.kapital {text-transform: capitalize}
</STYLE>
</HEAD>
<BODY>
<p class="besar">
pengubahan menjadi huruf Besar (Kapital)
</p>
<p class="kecil">
PENGUBAHAN MENJADI HURUF KECIL
```

```
</p>
<p class="kapital">
huruf kapital pada setiap awal kata
</p>
</BODY>
</HTML>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut,



Gambar 4.6 penerapan CSS untuk pengaturan text-transform

7) Pemberian efek bayangan pada teks/Text shadow

Text-shadow digunakan untuk memberikan efek bayang pada sebuah teks.

Berikut adalah format penulisan CSS dengan text-shadow

```
Selector { text-shadow: koordinatX koordinatY
nilaiWarnaTeksBayangan;}
```

Contoh : penulisan text shadow

```
h1 { text-shadow: 2px 2px #ff0000; }
```

Berikut ini adalah contoh penerapan dari style tersebut :

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {text-shadow:2px 3px #FF0000;}
</style>
</head>
```

```
<body>
<h1>Efek Text-shadow </h1>
<p><b>Catatan:</b>  Internet Explorer Versi 9 dan
sebelumnya tidak support pada text-shadow property.</p>
</body>
</html>
```

Bila listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 4.7 penerapan CSS untuk pemberian efek bayangan

8) Pengaturan Text Alignment

Properti text-align digunakan untuk mengatur alignment horizontal teks. Teks dapat ditulis rata kiri, rata kanan, rata kanan kiri ataupun ditulis center. Ketika text-align diset “justify", setiap baris memiliki lebar yang sama, dan margin kiri dan kanan lurus (seperti di majalah dan surat kabar). Format penulisan CSS untuk text alignment

```
selector {text-align:nilai text-align;}
```

Berikut adalah contoh dari penulisan style text-align yang disisipkan di selector h1 dan paragraf.

```
h1 {text-align:center;}
p.tanggal {text-align:right;}
p.utama {text-align:justify;}
```

pada selector p.tanggal dan p.utama artinya CSS akan diberikan ke paragraf dengan pemanggilan kelas “tanggal” dan “utama”. Untuk lebih lengkapnya dapat dilihat pada contoh berikut :

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {text-align:center;}
p.tanggal {text-align:right;}
p.utama {text-align:justify;}
</style>
</head>
<body>
<h1>Contoh CSS text-align</h1>
<p class="tanggal">07 Desember 2013</p>
<p class="utama">Properti text-align digunakan untuk
mengatur alignment horizontal teks.
Teks  dapat ditulis rata kiri, rata kanan, rata kanan
kiri ataupun ditulis center. Ketika text-align diset
“justify", setiap baris memiliki lebar yang sama, dan
margin kiri dan kanan lurus (seperti di majalah dan surat
kabar).
Format penulisan CSS untuk text alignment.selector {text-
align:nilai text-align;}
Berikut adalah contoh dari penulisan style text-align
yang disisipkan di selector h1 dan paragraf.
h1 {text-align:center;}
p.tanggal {text-align:right;}
p.utama {text-align:justify;}
pada selector p.tanggal dan p.utama artinya CSS akan
diberikan ke paragraf dengan pemanggilan kelas “tanggal”
dan “utama”. Untuk lebih lengkapnya dapat dilihat pada
contoh berikut :</p>
```

```
<p><b>Catatan:</b> Baris ini adalah contoh penulisan tanpa
CSS</p>
</body>
</html>
```

Bila listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 4.8 penerapan CSS untuk pengaturan text-align

c. Rangkuman.

Dari paparan materi pembahasan kegiatan belajar menyajikan cascading style sheet untuk memformat tampilan teks dapat dibuat rangkuman materi sebagai berikut :

- Agar lebih menarik, mudah dalam hal pengaturans serta editing, teks pada sebuah halaman web dapat diberikan sytle dalam kemasan CSS.
- Pengaturan style tersebut meliputi banyak properti, diantaranya color, direction, letter-spacing,line-height, text-align,text-decoration, text-indent, text-shadow, text-transform, unicode-bidi, vertical-align,white-space, word-spacing
- Format penulisan untuk pemberian warna pada text menggunakan CSS adalah sebagai berikut : selector {color:nilai warna}
- Format penulisan CSS untuk pengaturan jarak (spasi) antar karakter atau huruf adalah sebagai berikut selector { letter-spacing:nilai spasi;}
- Format penulisan CSS untuk mengaturan dekorasi teks adalah sebagai berikut selector {text-decoration : nilai text-decoration }
- Format penulisan CSS untuk pengaturan jarak (spasi) antar karakter atau huruf adalah sebagai berikut selector { word-spacing:nilai spasi;}

- Format penulisan untuk pengaturan jarak indentasi text adalah sebagai berikut selector { text-indent:nilai indent}
- . Format penulisan CSS untuk pengaturan text indent adalah sebagai berikut selector { text-transform:nilai text transform}
- format penulisan CSS dengan text-shadow adalah
  Selector {text-shadow: koordinatX koordinatY nilaiWarnaTeksBayangan ;}
- Format penulisan CSS untuk text alignment selector {text-align:nilai text-align;}

d. Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1. Buatlah tampilan halaman web dengan memberikan style pada teks yang digabungkan dengan tabel serta gambar seperti pada contoh berikut



2. Buatlah tampilan teks dalam fielset dengan memberikan style pada teksnya seperti pada gambar berikut :



3. Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e. Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Sebutkan sedikitnya 5 properti dari pengaturan teks menggunakan CSS
2. Tuliskan format penulisan CSS untuk memberikan warna pada teks
3. Tuliskan format penulisan CSS untuk memberikan efek bayangan
4. Tuliskan format penulisan CSS untuk pengaturan spasi antar karakter pada paragraf
5. Tuliskan format penulisan CSS untuk pengaturan spasi antar kata pada paragraf

f. Lembar Jawaban Tes Formatif.

LJ- 01 : 5 properti teks

....

....

....

....

....

....

....

....

....

....

LJ- 02 : Format penulisan CSS untuk warna teks

....

....

....

....

....

....

....

....

....

....

LJ- 03 : Format penulisan CSS untuk efek bayangan teks

....

....

....

....

....

....

....

....

....

....

LJ- 04 : Format penulisan CSS untuk mengatur spasi antar karakter

....

....

....

....

....

....

....

....

....

LJ- 05 : Format penulisan CSS untuk mengatur spasi antar kata

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

5. Kegiatan Belajar 5 : Menyajikan Cascading Style Sheet untuk Memformat Tampilan Multimedia

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 5 diharapkan peserta didik dapat :

1) Memahami style-style dasar CSS untuk memformat gambar
2) Menerapkan beberapa style untuk memformat tampilan gambar
3) Memahami style-style dasar CSS untuk memformat video
4) Menerapkan beberapa style untuk memformat tampilan video
5) Menyajikan CSS untuk memformat tampilan multimedia pada sebuah halaman web

b. Uraian materi.

Cascading style sheet dapat pula diterapkan pada elemen-elemen HTML multimedia, diantaranya gambar,video,audio.

1) Cascading style sheet pada tampilan gambar

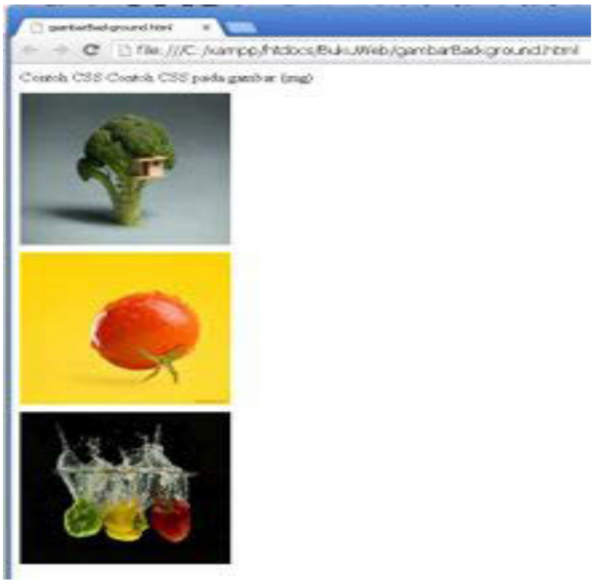
CSS berperan penting dalam menyajikan informasi dengan visual yang lebih baik, lebih optimal dari sisi penulisan serta lebih disenangi oleh mesin pencari. CSS dapat diterapkan hampir pada semua objek yang disisipkan dalam tag html baik itu tulisan, tabel dan gambar maupun objek lainya. Untuk mengatur gambar dapat dituliskan secara inline seperti contoh berikut :

```
<img src="nama-file-gambar" width="lebar" height="tinggi"
alt="text" title="text" border="1" />
```

Selain dengan cara inline , stlyle CSS dapat diberikan secara embed ataupun eksternal CSS. Berikut terdapat tiga gambar yang diberi style dimana ukuran ketiga gambar dibuat sama 200 x 200 dan margin atas gambar adalah 10px sehingga antar gambar terlihat ada jarak.

```
<! DOCTYPE HTML>
<html>
<head>
<tittle>Contoh CSS</tittle>
<style>
Img { Margin-top: 10px;
     Float: left;
     Clear: both;
     width:200px;
     height:200px;}
</style>
</head>
<body>
Contoh CSS pada gambar (img)<br>
<img src="brokoli.jpg" >
<img src="tomat.jpg">
<img src="paprika.jpg">
</body>
</html>
```

Bila listing kode diatas dijalankan maka akan menghasilkan tampilan gambar seperti berikut ini :



Gambar 5.1 penerapan CSS untuk image

2) Cascading style sheet Gambar untuk background

Seperti penambahan warna latar, penambahan gambar latar belakang juga dilakukan dengan menggunakan dua properti, yaitu background dan background-image. Nilai yang diisikan untuk properti ini ialah url dari gambar, relatif terhadap file CSS, sama seperti pada penambahan font.

```
div { background: url(“latar.png”);
background-image: url(“latar.png”);}
```

kesalahan tersebut bekerja, simpan dan jalankan kode Namun, penambahan gambar dengan cara di atas akan memberikan masalah ketika ukuran gambar tidak sama dengan ukuran elemen. Gambar akan ditampilkan berulangkali, secara horizontal maupun vertikal, jika ukuran elemen lebih besar dari ukuran gambar. Untuk dapat melihat bagaimana berikut :

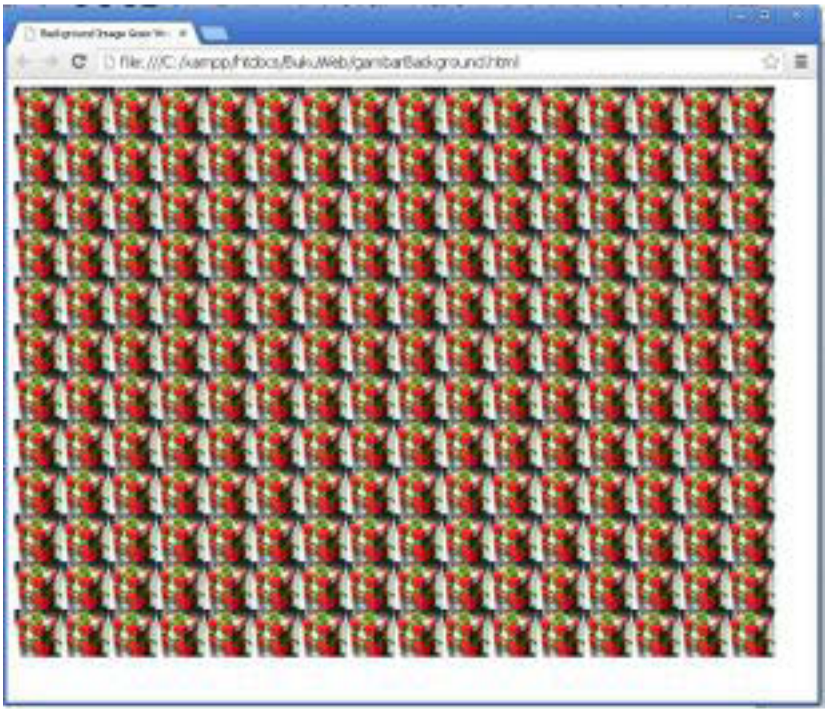
```
<html>
<head>
<title>Background Image Goes Wrong</title>
<style type="text/css">
#utama{background-
image:url(kiwi.jpg);height:600px;width:800px;}
</style>
</head>
<body>
<div id="utama">
</div>
</body>
</html>
```

Gambar asli yang digunakan sebagai background dengan ukuran 50px X 50px



Gambar 5.2 kiwi.jpg

Karena ukuran asli gambar adalah 50px X 50px sedangkan ukuran background yang diinginkan adalah 800px X 600px maka gambar akan ditampilkan berulang-ulang sampai terpenuhi ukuran background 800px X 600px seperti pada gambar berikut :



Gambar 5.3 penerapan CSS untuk image background

Adakalanya perulangan tampilan gambar otomatis ini tidak diinginkan. Untuk itu dapat menggunakan properti background-repeat untuk mengatur bagaimana perulangan kode ingin dilakukan. Terdapat empat nilai yang dapat diisikan pada properti background-repeat, yaitu repeat, no-repeat, repeat-x, dan repeat-y. Seperti namanya, no-repeat menghilangkan perulangan, repeat-x memberikan perulangan hanya pada sumbu x (horizontal), dan repeat-y memberikan perulangan hanya pada sumbu y (vertikal). Listing Kode CSS di bawah akan menghilangkan perulangan pada sumbu x dan sumbu y, sehingga kiwi,jpg hanya akan ditampilkan satu kali :

```
#utama {background-image: url(kiwi.jpg);
     background-repeat: no-repeat;
     height: 600px;width: 800px;}
```

3) Cascading style sheet untuk menyisipkan gambar

Adakala saat membuat halaman web yang berisi artikel diperlukan gambar yang mendukung artikel tersebut. Agar terlihat rapi dan menarik maka gambar yang disisipka perlu diberi CSS.

```
<html>
<head>
<title>
CSS image placement
</title>
<style >
 #headline1 { background-image: url(kiwi.jpg);
              background-repeat: no-repeat;
              background-position: left top;
              padding-top:68px;
             margin-bottom:50px; }
 #headline2 { background-image: url(kiwi.jpg);
              background-repeat: no-repeat;
              background-position: left top;
              padding-top:68px; }
</style>
</head>
<body>
<div id="headline1">CSS (versi sekarang adalah CSS3)
banyak dilibatkan dalam dokumen Web.
 Kegunaannya adalah untuk memformat dokumen.
 Sebagai contoh, warna teks atau bahkan warna
latarbelakang bisa diatur melalui CSS.
</div>
<div id="headline2">Namun, tentu saja kegunaan CSS jauh
lebih
```

Bila listing kode dijalankan akan menghasilkan tampilan sebagai berikut :



Gambar 5.4 penerapan CSS untuk penyisipan image pada teks

4) Cascading style sheet pada tampilan video

Untuk penerapan Cascading style shee pada elemen video dapat dituliskan secara inline seperti contih berikut ini :

```
<html>
<head>
<title>Test</title>
</head>
<body>
<video controls="controls" style="display:block; margin:
0 auto; width:400px ; heigth:400;" >
<source src="movie.mp4" type="video/mp4" />
Your browser does not support the video tag.
</video>
</body>
```

Bila listing program diatas dijalankan akan menghasilkan tampilan video yang di atur tampil “center” dengan atribut width:400px dan height:400px seperti pada tampilan berikut ini :



Gambar 5.5 penerapan css pada video

5) Cascading style sheet video embed dari youtube

Penyajian video juga bisa memanfaatkan elemen iframe yang dilinkkan ke alamat video tertentu , misalnya www.youtube.com. Berikut adalah penerapan CSS secara embed untuk menampilan video dari youtube.

```
<html>
<head>
<title>Test</title>
<style>
.videoWrapper { position: relative;
                padding-bottom: 56.25%; /* 16:9 */
               padding-top: 25px;
               height: 0; }
.videoWrapper iframe { position: absolute;
                       top: 0;
                       left: 0;
                       width: 100%;
                       height: 100%; }
</style>
</head>
<body>
```

```
<div class="videoWrapper">
<!-- Copy & Pasted from YouTube -->
<iframe width="560" height="349"
src="http://www.youtube.com/embed/n_dZNLr2cME?rel=0&hd=1"
frameborder="0" allowfullscreen></iframe>
</div>
</body>
</html>
```

Bila listing kode diatas dijalankan dalam keadaan komputer terhubung dengan internet akan menghasilkan tampilan sebagai berikut :



Gambar 5.6 video dengan embed youtube

c. Rangkuman

Dari paparan materi pembahasan kegiatan belajar menyajikan cascading style sheet untuk memformat tampilan multimedia dapat dibuat rangkuman materi sebagai berikut :

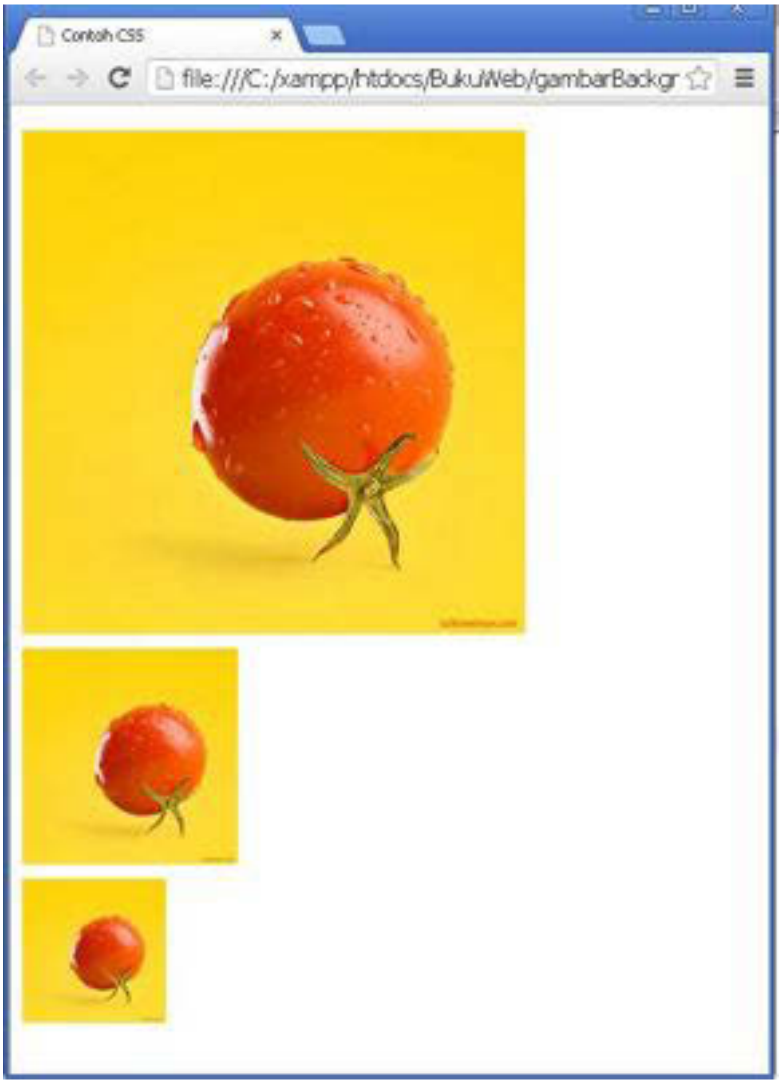
- CSS berperan penting dalam menyajikan informasi dengan visual yang lebih baik, lebih optimal dari sisi penulisan serta lebih disenangi oleh mesin pencari.
- Untuk mengatur format multimedia dapat dituliskan secara inline, embed maupun eksternal CSS.
- Pemberian style pada gambar dapat menggunakan selector <img>
- Pemberian style pada video dapat menggunakan selector <video>
- Penambahan gambar sebagai latar belakang menggunakan dua properti, yaitu property background image dengan menggunakan tag div seperti format berikut

```
div { background-image: url("latar.png"); }
```

d. Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1. Buatlah tampilan halaman web yang memanggil video dari youtube , sebelum video muncul ada image yang ditampilkan terlebih dahulu. Posisi dari video ada di sebelah kanan halaman web
2. Buatlah tampilan halaman web yang menampilkan 3 gambar dengan ukuran yang berbeda



3. Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e.Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Tuliskan format penulisan CSS untuk gambar menggunakan tag <img>
2. Tuliskan format penulisan CSS untuk gambar menggunakan tag <div>
3. Tuliskan format penulisan CSS untuk video menggunakan tag <video>

f. Lembar Jawaban Tes Formatif.

LJ- 01 : CSS untuk gambar menggunakan tag <img>

....

....

....

....

....

....

....

....

....

....

LJ- 02 : CSS untuk gambar menggunakan tag <div>

....

....

....

....

....

....

....

....

....

....

LJ- 03 : CSS untuk video menggunakan tag <video>

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

6. Kegiatan Belajar 6 : Penerapan CSS pada elemen tabel

a. Tujuan Pembelajaran.

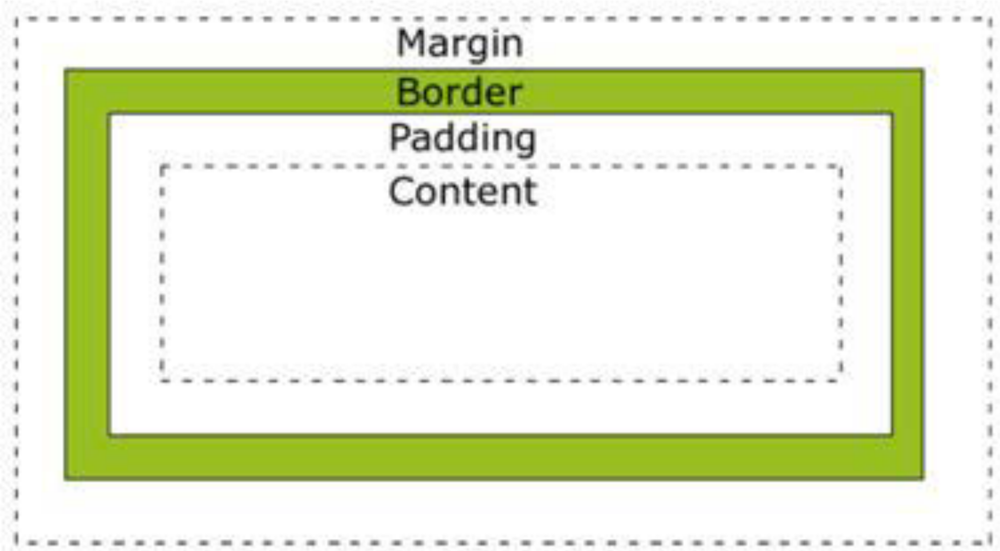
Setelah mengikuti kegiatan belajar 6 diharapkan peserta didik dapat :

1) Memahami style-style dasar CSS untuk memformat tampilan tabel
2) Menerapkan beberapa style untuk memformat tampilan tabel
3) Menyajikan CSS untuk memformat tampilan tabel pada sebuah halaman web

b. Uraian materi.

1) Konsep dasar model box

Pada dasarnya elemen dari HTML adalah model box. Istilah model box ini digunakan pada saat membahas desain dan layout. Model box memungkinkan untuk membuat border disekeliling elemen dan space elemen yang saling berhubungan. Model box pada elemen HTML dapat digambar sebagai berikut :



Gambar 6.1 konsep model box

Keterangan gambar

Margin	**Daerah paling diluar yang membatasi dengan elemen HTML yang lainnya dan tidak berwarna**
Border	Border mengelilingi padding dan content. Garis yang mempunyai warna dan ketebalan

Padding	Area yang mengelilingi content. Warna padding dipengaruhu oleh background dari box
Content	Isi dari box, bisa berupa teks maupun gambar

Ketika menentukan width dan height dari properti elemen dengan menerapkan CSS, sebenarnya itu hanya mendefiniskan area content saja sehingga jika menginginkan keseluruhan eleme box maka harus diperhitungkan juga margin-border-padding

2) Pengaturan border tabel

CSS dapat digunakan untuk memformat border atau garis tepi dari sebuah tabel. Format yang dapat dilakukan meliputi jenis garis (dotted, solid, dan lain-lain), warna border, ketebalan garis tepi dan sebagainya. Demikian pula warna dari sebuah sel, dapat diatur dengan CSS.

Berikut adalah contoh penerapan style border pada tabel dengan style sebagai berikut {border:1px solid black;} dan listing kode lengkapnya seperti dibawah ini

```
<html>
<head>
<style>
table,th,td
{ border:1px solid black; }
</style>
</head>
<body>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</tr>
<tr>
<td>Adelin</td>
<td>Larasati</td>
```

```
</tr>
<tr>
<td>Novero</td>
<td>Dayandra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html>
```

Jika listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 6.2 penerapan style border pada tabel

Berikut ini contoh pengaturan property

```
border-collapse:collapse;
```

yang menghasilkan tampilan garis single pada tabel

```
<html>
<head>
<style>
table
{ border-collapse:collapse; }
```

```
table, td, th
{ border:1px solid black; }
</style>
</head>
<body>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</tr>
<tr>
<td>Adelin</td>
<td>Larasati</td>
</tr>
<tr>
<td>Novero</td>
<td>Dayandra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html>
```

Jika listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 6.3 penerapan style border-collapse pada tabel

Pada saat memberikan style border → {border:1px solid black;}

Hal ini berarti seluruh border akan efek terkena 1px solid black; baik bagian atas (top),bagian bawah(bottom), bagian kiri(left) ataupun kanan (right). Border dapat dikenai style untuk area tertentu saja karena border dapat diseting untuk border bawah atas kanan dan kiri seperti pada style berikut :

```
<style>
table,th,td
{ border-top: 2px solid red;
  border-right: 2px solid green;
  border-left: 2px solid blue;
  border-bottom: 2px solid #DD4A04; }
</style>
```

3) Pengaturan lebar dan tinggi tabel

Lebar dan tinggi dari sebuah tabel didefinisikan dalam property width dan height Contoh berikut ini mengatur lebar tabel 100% dan tinggi dari elemen th adalah 50px.

```
table
{width:100%; }
th
{ height:50px; }
```

Berikut ini penerapan style width dan height dalam sebuah tabel

```
<html>
<head>
<style>
table,td,th
{ border:1px solid black; }
table
{ width:100%; }
th
{ height:50px; }
```

```
</style>
</head>
<body>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</tr>
<tr>
<td>Adelin</td>
<td>Larasati</td>
</tr>
<tr>
<td>Novero</td>
<td>Dayandra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html>
```

Jika listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 6.4 penerapan style width dan height pada tabel

4) Pengaturan text-align pada tabel

Teks dalam tabel diatur dengan properti.text-align dan vertical-align . Nilai dari properti text-align untuk horizontal alignment adalah left, right, atau center.

```
Td
{ text-align:right; }
```

Nilai dari properti text-align untuk horizontal alignment adalah top, bottom, or middle.

```
Td
{ height:50px;vertical-align:bottom; }
```

Berikut ini penerapan style text-align dalam sebuah tabel, dimana teks diatur dengan posisi rata kanan

```
<html>
<head>
<style>
table,td,th
{ border:1px solid black; }
td
{ <body> text-align:right;height:50px;
vertical-align:bottom;}
</style>
</head>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</tr>
<tr>
<td>Adelin</td>
<td>Larasati</td>
</tr>
```

```
<tr>
<td>Novero</td>
<td>Dayandra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html>
```

Jika listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 6.5 penerapan style text-align pada tabel

5) Pengaturan padding tabel

Padding adalah jarak antara isi atau content, berupa teks ataupun gambar dengan batas dalam dari sel tersebut. Style padding ini dapat diterapkan pada elemen HTML <td> (kolom). Ukuran yang digunakan dapat berupa cm, atau px (piksel). Property style untuk pengaturan padding yang mengelilingi content adalah td{ padding:15px; }

Padding dapat pula diatur untuk bagian tertentu saja, misalnya adalah :

- padding-top □jarak dari atas sel
- padding-bottom □jarak dari bawah sel
- padding-left □jarak dari kiri sel

- padding-right jarak dari kanan sel.

Berikut ini adalah beberapa contoh penggunaan style padding pada tabel :

```
<html>
<head>
<style>
table,td,th
{ border:1px solid black; }
td
{ text-align:right;height:50px;vertical-align:bottom;
padding:15px; }
</style>
</head>
<body>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</td>
</tr>
<tr>
<td>Novero</td>
<td>Dayand</tr>
<tr>
<td>Adelin</td>
<td>Larasati ra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html>
```

listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



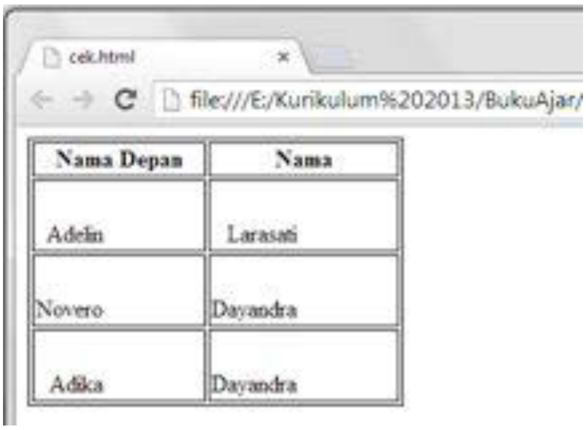
Gambar 6.6 penerapan style padding pada tabel

Jika menginginkan padding diarea tertentu saja, misalnya padding untuk area kanan maka dapat diterap padding-right

```
<html>
<head>
<style>
table,td,th
{ border:1px solid black; }
td
{ text-align:right;height:50px;vertical-align:bottom;
padding-right: 70px; }
</style>
</head>
<body>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</tr>
<tr>
<td>Adelin</td>
<td>Larasati</td>
</tr>
```

```
<tr>
<td>Novero</td>
<td>Dayandra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html
```

Jika listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :



Gambar 6.7 penerapan style padding-right pada tabel

6) Pengaturan warna pada table

Pewarnaan pada tabel dapat digunakan untuk mengatur warna pada teks,background maupun border.

```
table, td, th
{ border:1px solid green; }
```

Style diatas akan memberikan efek warna green pada border table,data (td) dan header tabel (th). Style warna juga bisa diberikan hanya pada bagian tertentu saja, misalnya bagian header tabel (th) seperti pada listing css dibawah

```
Th
{ background-color:green;
color:white; }
```

Header tabel akan berwarna green dan teksnya berwarna putih. Penerapan style diatas pada sebuah tabel akan terlihat pada contoh berikut :

```
<html>
<head>
<style>
table, td, th
{ border:1px solid green; }
th
{ background-color:green;color:white;}
</style>
</head>
<body>
<table>
<tr>
<th>Nama Depan</th>
<th>Nama</th>
</tr>
<tr>
<td>Adelin</td>
<td>Larasati</td>
</tr>
<tr>
<td>Novero</td>
<td>Dayandra</td>
</tr>
<tr>
<td>Adika</td>
<td>Dayandra</td>
</tr>
</table>
</body>
</html>
```

Jika listing kode diatas dijalankan maka akan menghasilkan tampilan sebagai berikut :

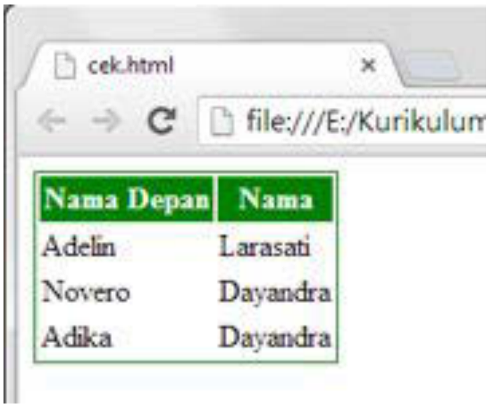


Gambar 6.8 penerapan style color pada tabel

Ketika pada td tidak diberi style seperti pada style berikut :

```
<style>
table,th
{ border:1px solid green; }
th
{ background-color:green;color:white; }
</style>
```

Akan menghasilkan tampilan tabel seperti berikut :



Gambar 6.9 td tidak diberi style tersendiri

c. Rangkuman.

Dari paparan materi pembahasan kegiatan belajar menyajikan cascading style sheet untuk memformat tampilan tabel dapat dibuat rangkuman materi sebagai berikut :

- Pada dasarnya elemen dari HTML adalah model box yang mempunyai properti margin,border,padding dan content hal ini dapat digunakan untuk mendesain tampilan dari tabel.
- CSS dapat digunakan untuk memformat border atau garis tepi dari sebuah tabel.
- penerapan style border pada tabel dengan style sebagai berikut {border:1px solid black;}
- Lebar dan tinggi dari sebuah tabel didefinisikan dalam property width dan height. Teks dalam tabel diatur dengan properti.text-align dan vertical- align .
- Nilai dari properti text-align untuk horizontal alignment adalah left, right, atau center.
- Nilai dari properti text-align untuk horizontal alignment adalah top, bottom, or middle
- Padding adalah jarak antara isi atau content, berupa teks ataupun gambar dengan batas dalam dari atau elemen HTML <td>.
- Pewarnaan pada tabel dapat digunakan untuk mengatur warna pada teks,background maupun border
- Style warna akan memberikan efek warna pada tag table,data (td) dan header tabel (th).

d. Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1.Buatlah tabel seperti berikut dengan mengatur warna di elemen tag <th>

Nama Computer	Prossesor	Ram	System Operasi
Clnt-1	IP 4 1,8 Ghz	10 GB	Redhat Linux
Clnt-2	IP 2 Ghz C	6 GB	Mandrake Linux
Clnt-2	XP 2000+	20 GB	Windows

2. Buatlah tabel seperti berikut dengan

Kompetensi Keahlian	Kode	Jml Diklat	Jml Kaitan	Pencapaian (%)
Animasi	TIKAA073	8	0	0
Teknik Komputer dan Jaringan	TIKKJ071	21	21	100
Teknik Multimedia	TIKMM072	10	0	0
Rekayasa Perangkat Lunak	TIKPL070	21	17	81
Teknik Jaringan Akses	TITJA069	0	0	0

Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e. Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Jelaskan tentang model box pada elemen html
2. Tuliskan format penulisan CSS untuk pengaturan border pada tabel
3. Tuliskan format penulisan CSS untuk pengaturan warna pada tabel
4. Tuliskan format penulisan CSS untuk pengaturan teks pada tabel
5. Tuliskan format penulisan CSS untuk pengaturan lebar dan tinggi tabel

f. Lembar Jawaban Tes Formatif.

LJ- 01 : model box pada elemen html

....

....

....

....

....

....

....

....

....

....

LJ- 02 : format penulisan CSS untuk pengaturan border pada tabel

....

....

....

....

....

....

....

....

....

....

LJ- 03 : format penulisan CSS untuk pengaturan warna pada tabel

....

....

....

....

....

....

....

....

....

....

LJ- 04 : format penulisan CSS untuk pengaturan teks pada tabel

....

....

....

....

....

....

....

....

....

LJ- 05 : format penulisan CSS untuk pengaturan lebar dan tinggi tabel

....

....

....

………

………

………

………

………

………

g. Lembar Kerja Siswa.

7. Kegiatan Belajar 7 : Menyajikan Cascading Style Sheet untuk Memformat Tampilan Form

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 8 diharapkan peserta didik dapat :

1) Memahami style CSS untuk memformat tampilan form
2) Menerapkan beberapa style dasar CSS untuk memformat tampilan form

b. Uraian materi.

Tampilan sebuah form akan lebih terlihat menarik apabila diterapkan CSS di dalamnya. Style CSS dapat digunakan di berbagai komponen input di dalam form. Dengan menggunakan style CSS akan mengurangi beban halaman dan konsumsi bandwidth.

1) Cascading style sheet pada elemen form

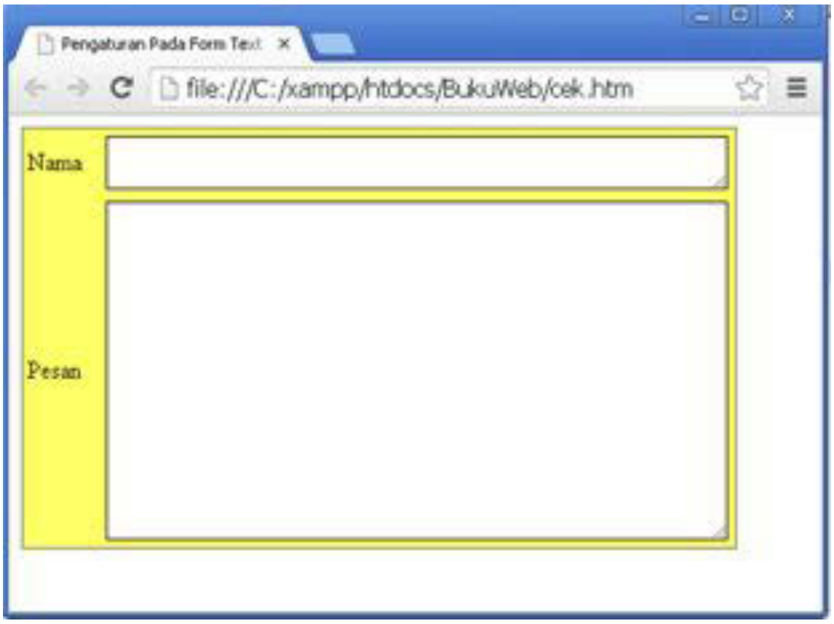
Agar form terlihat lebih indah menarik serta mudah dalam pengontrolan desain salah satunya memberikan style CSS. Dibawah ini adalah style yang bisa diberikan pada elemen form diantaranya pengaturan besar dan jenis border, warna border, lebar form, batas form dari sisi layout, dan warna background dari form. Sebenarnya masih banyak style lain yang bisa diberikan ke form

```
Form
{ border:1px solid #666666;
  width : 480; /*lebar form*/
  margin-left:0; /*jarak dari batas kiri layout*/
  background-color:#ffff66;}
```

Bila style form diatas di embedkan ke dalam file html akan listing kodenya seperti berikut ini :

```
<HTML>
<HEAD>
<TITLE>Pengaturan Pada Form Text Area</TITLE>
<STYLE ="text/css">
form{ border:1px solid #666666;
      width : 70%;
      margin-left:0; /*jarak fieldset dari batas kiri
       layout*/
      background-color:#ffff66;}
</STYLE>
</HEAD>
<BODY>
<form name="form1" method="post" action="">
<table><TR>
<TD width="50" >Nama </TD>
<TD width="175" >
          <textarea name="textarea"
cols=50></textarea></TD>
</TR>
     <TR>
<TD width="50" >Pesan </TD>
<TD width="175" >
          <textarea name="textarea"cols=50 rows=14>
</textarea></TD>
</TR></table>
</form>
</BODY>
</HTML>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut



Gambar 7.1 form dengan css

2) Cascading style sheet pada input text

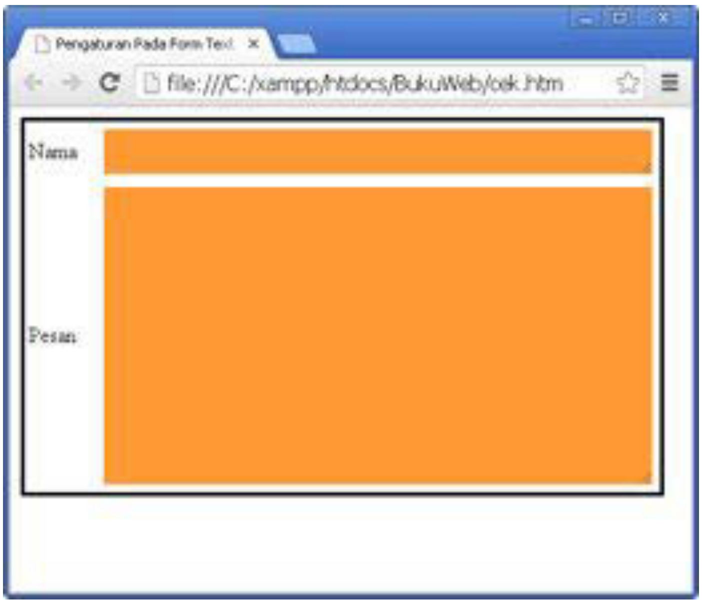
Textarea mendefinisikan kontrol input berupa area teks pada form .Elemen form textarea dapat diatur stylenya dengan menggunakan selector textarea `{…}`. Berikut contoh penerapan CSS untuk elemen textarea.

```
<HTML>
<HEAD>
<TITLE>Pengaturan Pada Form Text Area</TITLE>
<STYLE ="text/css">

    TEXTAREA
    { COLOR : white ;background-color : #FF9933 ; }
</STYLE>
</HEAD>
<BODY>
<form name="form1" method="post" action="">
<TABLE bordercolor="1">
<TR>
<TD width="50" >Nama </TD>
<TD width="175" >
```

```
            <textarea name="textarea"
cols=50></textarea></TD>
</TR>
     <TR>
<TD width="50" >Pesan </TD>
<TD width="175" >
            <textarea name="textarea"cols=50 rows=14 >
</textarea></TD>
</TR>
</TABLE>
</form>
</BODY>
</HTML>
```

Bila listing kode diatas dijalankan akan menghasilkan tampilan seperti berikut.



Gambar 7.2 penerapan css pada textarea

3) Cascading style sheet pada text field

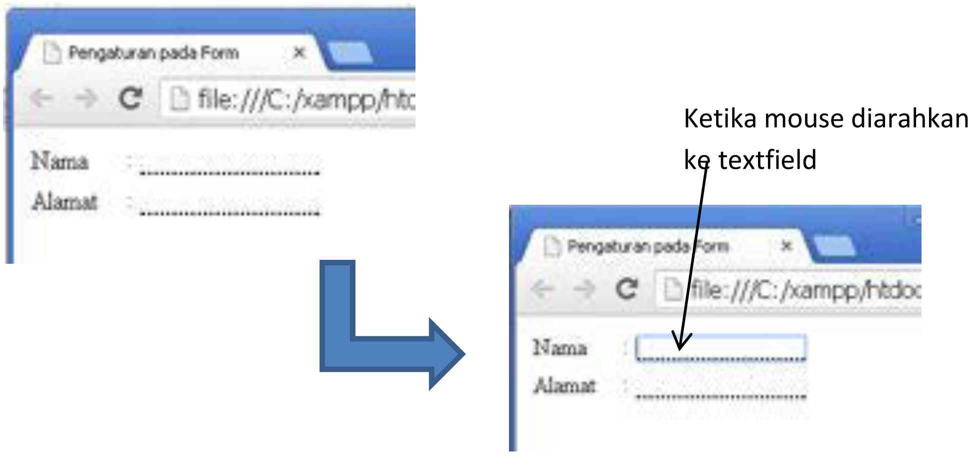
extfield mendefinisikan kontrol input text pada form. Berikut ini adalah contoh format CSS pada input yang membuat efek dari input textfield menjadi seperti

sisi formulir cetakan. CSS untuk textfield menggunakan selector input. Style yang diberikan pada input textfield diantaranya warna, background, border, jenis serta ukuran font dan masih banyak atribut lainya yang bisa digunakan untuk style textfield.

```
<HTML>
<HEAD>
<TITLE>Pengaturan pada Form</TITLE>
<STYLE ="text/css">
    input {color:#000000;
           background:white;
           border-color:white;
           border-bottom-color:black;
           border-top:0px solid;
           border-bottom:2px dotted;
           border-left: 0px solid;
           border-right:0px solid;
           font-family:tahoma,Arial;
           font-size:11px;}
</STYLE>
</HEAD>
<BODY>
<form name="form1" method="post" action="">
<TABLE>
<TR>
<TD width="58" >Nama </TD>
<TD width="230" >:
<input type="text" name="nama">
    </TD>
</TR>
<TR>
<TD width="58">Alamat</TD>
<TD width="230" >:
<input type="text" name="alamat">
    </TD>
```

```
</TR>
</TABLE>
</form>
</BODY>
</HTML>
```

Bila listing kode dijalankan akan menampilkan tampilan berikut ini :



Gambar 7.3 penerapan css pada textfield

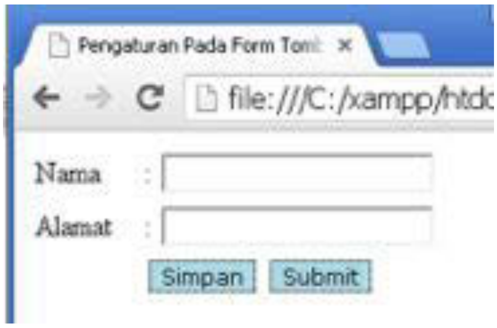
4) Cascading style sheet pada button

Button merupakan tombol yang dapat diklik. Pada tombol, baik submit, reset maupun button, style CSS dapat digunakan untuk menimbulkan efek pipih (flat) pada tombol. Berikut ini adalah contoh penggunaan CSS pada tombol flat :

```
<html>
<head>
<title>pengaturan pada form tombol</title>
<style ="text/css">
    input.button { color:#000000;
                   background: lightblue ;
                   border-color:white;
                   border-bottom-color:black;
                   border-top:1px dotted;
                   border-bottom:1px dotted;
                   border-left: 1px dotted;
                   border-right:1px dotted;
                   font-family:verdana;
```

```
                    font-size:12px; }
</style>
</head>
<body>
<form name="form1" method="post" action="">
<table >
<tr>
<td width="58" >nama </td>
<td width="230" >:
<input type="text" name="nama">
          </td>
</tr>
<tr>
<td width="58">alamat</td>
<td width="230" >:
<input type="text" name="alamat">
          </td>
</tr>
<tr>
<td></td>
<td ><input  class="button" type="submit" name="submit"
value="simpan">
<input   class="button"   type="submit"   name="submit2"
value="submit">
          </td>
</tr>
</table>
</form>
</body>
</html>
```

Bila listing kode diatas dijalankan akan menampilkan style button dengan efek flat/pipih seperti berikut ini :



Gambar 7.4 style button flat

Efek style juga bisa diberikan saat button dilewati mouse. Untuk memberikan efek berbeda saat button dilewati mouse dapat menggunakan selector CSS input.button:hover . Untuk listing kode lengkapnya adalah sebagai berikut :

```
<HTML>
<HEAD>
<TITLE>Pengaturan Pada Form Tombol</TITLE>
<STYLE ="text/css">
input.button
    { color: #fff;
      background: #ffa20f;
      border: 2px outset #BBD16D;
      font-family: "calibri", Times, serif;
      font-size: 14px;
      font-weight: bold;
      width : auto; }
input.button:hover
    { color: #fff;
      background: #0c62a4; /*biru kayak judul header*/
      border: 2px outset #ffa20f;
      font-family: "calibri", Times, serif;
      font-size: 14px;
      font-weight: bold;}
</STYLE>
</HEAD>
<BODY>
```

```
<form name="form1" method="post" action="">
<TABLE >
<TR>
<TD width="58" >Nama </TD>
<TD width="230" >:
<input type="text" name="nama">
          </TD>
</TR>
<TR>
<TD width="58">Alamat</TD>
<TD width="230" >:
<input type="text" name="alamat">
          </TD>
</TR>
<TR>
<TD></TD>
<TD ><input  class="button" type="submit" name="Submit"
value="Simpan">
<input class="button" type="submit" name="Submit2"
value="Submit">
          </TD>
</TR>
</TABLE>
</form>
</BODY>
</HTML>
```

Bila listing tersebut dijalankan akan muncul tampilan form dengan keadaan yang berbeda saat mouse berada melewatinya dan tidak melewatinya. Dibawah ini keadaan button ketika mouse tidak berada diatasnya. Style button yang bekerja saat mouse tidak melewati button

```
input.button
     { color: #fff;
```

```
        background: #ffa20f;
        border: 2px outset #BBD16D;
        font-family: "calibri", Times, serif;
        font-size: 14px;
        font-weight: bold;
        width : auto;}
```

Sehingga muncul tampilan sebagai berikut :



Gambar 7.5 style form input.button

Saat mouse melewatinya, button yang semula berwarna orange berubah menjadi biru. Style untuk memberikan perubahan saat mouse melewati button adalah

```
input.button:hover
      { color: #fff;
        background: #0c62a4; /*biru kayak judul header*/
        border: 2px outset #ffa20f;
        font-family: "calibri", Times, serif;
        font-size: 14px;
        font-weight: bold;}
```

Sehingga menghasilkan efek sebagai berikut :



Gambar 7.6 style form input.button:hover

5) Cascading style sheet padaradio button

Radio button merupakan bagian dari elemen input form. Saat ingin memberikan style pada radio button dapat menggunakan selector input.selector. Contoh berikut adalah style CSS pada input yang bertipe pilihan radiobutton :

```
<HTML>
<HEAD>
<TITLE>Pengaturan pada Form</TITLE>
<STYLE ="text/css">
     INPUT.rb1{ background: blue ;
                border-color:white;
                font-family:verdana;
                font-size:12px; }
     INPUT.rb2{ background: green ;
                border-color:white;
                font-family:verdana;
                font-size:12px;}
</STYLE>
</HEAD>
<BODY>
<form name="form1" method="post" action="">
<TABLE bordercolor="1">
<TR>
<TD width="58" >Kelamin</TD>
<TD width="230" >:
<input type="radio" name="radiobutton"
value="radiobutton" class="rb1">
        Pria /Male</TD>
</TR>
<TR>
<TD width="58"> </TD>
<TD width="230" > 
<input type="radio" name="radiobutton"
value="radiobutton" class="rb2">
```

```
        Wanita/Famale </TD>
</TR>
</TABLE>
</form>
</BODY>
</HTML>
```

Untuk menerapkan style input.rd1 dan input.rd2 maka pada tag <input type="radio" > perlu memanggil style tersebut dengan cara class=”rd1” dan class=”rd2” . Sehingga ketika listing kode diatas dijalankan akan muncul tampilan radio button seperti berikut ini :



Gambar 7.7 penerapan style pada radio button

6) Cascading style sheet pada chexbox

Checkbox merupakan bagian dari elemen input form. Saat ingin memberikan style pada Checkbox dapat menggunakan selector input.selector. Contoh berikut adalah style CSS pada input yang bertipe pilihan Checkbox.

```
 <html>
<head>
<title>pengaturan pada form checkbox</title>
<style ="text/css">
    input.checkbox1{  background:  blue  ;border-color: green;
    border-style : outset double; }
    input.checkbox2{  background:  yellow  ;border-color: red;
    border-style :double; }
```

```
     input.checkbox3{     background:     green     ;border-
color:black;
     border-style :dotted; }
 </style>
 </head>
 <body>
 <form name="form1" method="post" action="">
 <table bordercolor="1">
 <tr>
 <td width="58" >hoby</td>
 <td width="230" ><input type="checkbox" name="badminton"
 class="checkbox1">
         badminton</td>
 </tr>
 <tr>
 <td width="58"> </td>
 <td width="230" ><input type="checkbox" name="tenismeja"
 class="checkbox2">
         tenis meja</td>
 </tr>
 <tr>
 <td> </td>
 <td ><input type="checkbox" name="sepakbola"
 class="checkbox3">
         sepakbola </td>
 </tr>
     <tr>
 <td> </td>
 <td ><input type="checkbox" name="golf"
 class="checkbox3">
       golf   </td>
 </tr>
 </table>
 </form>
```

```
</body>
</html>
```

Bila listing kode diatas dijalankan akan muncul tampilan checkbox seperti berikut :



Gambar 7.8 penerapan style pada check box

c. Rangkuman.

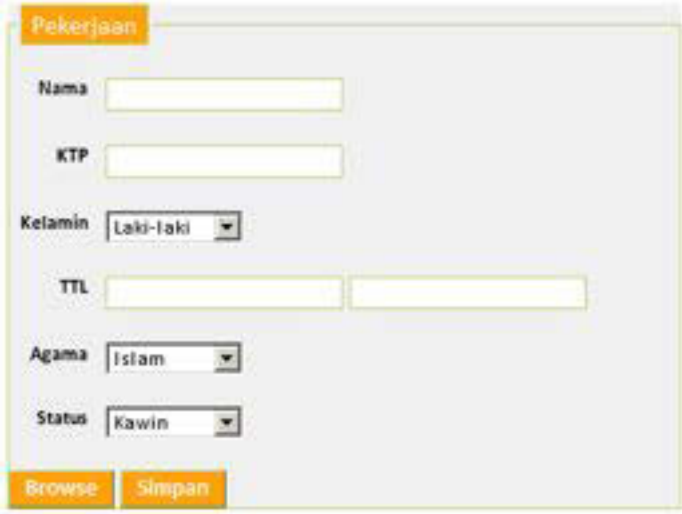
Dari paparan materi pembahasan kegiatan belajar menyajikan cascading style sheet untuk memformat tampilan formdapat dibuat rangkuman materi sebagai berikut :

- Agar form terlihat lebih indah menarik serta mudah dalam pengontrolan desain salah satunya memberikan style CSS.
- Dengan menggunakan style CSS akan mengurangi bebanhalamandankonsumsi bandwidth.
- style yang bisa diberikan pada elemen form diantaranya pengaturan besar dan jenis border,warna border, lebar form, batas form dari sisi layout, dan warna background dari elemen-elemen form.
- Elemen form dapat diatur stylenya dengan menggunakan selector form {…}.
- Elemen form textarea dapat diatur stylenya dengan menggunakan selector textarea {…}.
- Style yang diberikan pada input textfield diantaranya warna, background, border, jenis serta ukuran font dan masih banyak atribut lainya yang bisa digunakan untuk style textfield.

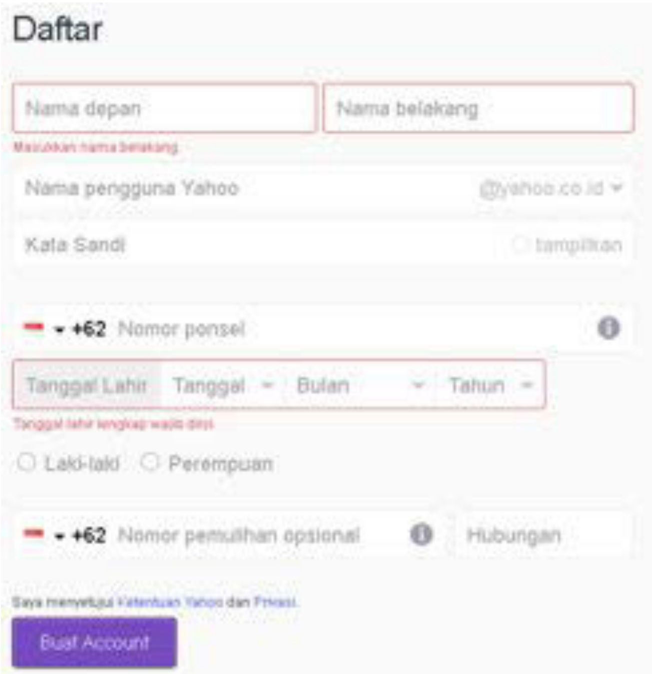
d.Tugas.

Sebelum mengerjakan tugas, buatlah kelompok. terdiri atas 2-3 orang. Dalam kegiatan ini masing-masing kelompok kerjakan tugas berikut :

1. Buatlah form dilengkapi CSS dengan menggunakan elemen fieldset, textfield serta select seperti contoh berikut ini



2. Buatlah form dilengkapi CSS dengan menggunakan elemen textfield,radio button serta select seperti contoh berikut ini



Kemudian secara bergantian masing-masing kelompok mempresentasikan hasilnya didepan kelas.

e. Tes Formatif.

Dalam test ini setiap peserta didik membaca dengan cermat dan teliti setiap butir soal dibawah ini. Kemudian berdasarkan uraian materi diatas tulislah jawabannya pada lembar jawaban test formatif yang telah disediakan.

1. Tuliskan format penulisan CSS untuk elemen form
2. Tuliskan format penulisan CSS untuk input text area
3. Tuliskan format penulisan CSS untuk text field
4. Tuliskan format penulisan CSS untuk radio button
5. Tuliskan format penulisan CSS untuk checkbox

f. Lembar Jawaban Tes Formatif.

LJ- 01 : CSS untuk elemen form

....

....

....

....

....

....

....

....

....

....

LJ- 02 : CSS untuk elemen textarea

....

....

....

....

....

....

....

....

....

....

LJ- 03 : CSS untuk elemen text field

....

....

....

....

....

....

....

....

....

....

LJ- 04 : CSS untuk elemen radio button

....

....

....

....

....

....

....

....

....

....

LJ- 05 : CSS untuk elemen checkbox

....

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

8. Kegiatan Belajar 8 : Pengenalan Javascript

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 8 ini siswa diharapkan dapat :

1) Mengetahui bahasa pemrograman pada halaman web klien
2) Memahami pembuatan dokumen javascript

b. Uraian Materi.

Pengenalan Javascript

Javascript adalah bahasa yang berbentuk kumpulan skrip yang berjalan pada suatu dokumen HTML. Sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengijinkan pengeksekusian perintah perintah di sisi user, yang artinya di sisi browser bukan di sisi server web.

Javascript diperkenalkan pertama kali oleh Brendan Eich yang bekerja di Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan “LiveScript” yang berfungsi sebagai bahasa sederhana untuk browser Netscape Navigator 2. Pada masa itu bahasa ini banyak di kritik karena kurang aman, pengembangannya yang terkesan buru buru dan tidak ada pesan kesalahan yang di tampilkan setiap kali kita membuat kesalahan pada saat menyusun suatu program. Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman “Java”) pada masa itu, maka Netscape memberikan nama “Javascript” kepada bahasa tersebut pada tanggal 4 desember 1995. Pada saat yang bersamaan Microsoft sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai “Jscript” di browser Internet Explorer 3. Versi terakhir dari bahasa ini adalah versi 1.8.5 yang dirilis tanggal 22 Maret 2011.

Oleh karena ada banyak perusahaan yang mengembangan, bahasa ini kemudian distandarkan dengan nama ECMAScript oleh Netscape melalui Organisasi Internasional ECMA. Standar ini dipublikasikan pertama kali pada bulan Juni 1997 dengan nama dokumen Spesifikasi ECMA-262. Melalui

publikasi tersebut, semua implementasi Javascript pada tiap browser akan memiliki standar penerapan (pengkodean) yang sama. Saat ini standar ini telah mencapai rilis Edisi ke-5.1, yang dipublikasikan pada bulan Juni 2011. Setiap browser saat ini memiliki implementasi sendiri-sendiri untuk ECMAScript ini, diantaranya Internet Explorer dengan JScript, Opera dengan ECMAScript, dan Mozilla Firefox, Google Chrome termasuk juga Safari dengan nama Javascript.

Selain di browser, sekarang Javascript juga sudah diterapkan pada banyak aplikasi lainnya seperti Windows 8 Apps (.Net Framework), Adobe Flash ActionScript, KDE Desktop Environment, Node.js, Qt QML, JQuery Mobile, Firefox OS, Ubuntu Touch dan masih banyak lagi kemungkinan implementasi lainnya.

Javascript bergantung kepada browser (navigator) yang memanggil dan menampilkan halaman web yang tidak hanya berisi HTML tapi juga dapat berisi skrip-skrip Javascript. Javascript juga tidakmemerlukan penterjemah khusus untuk menjalankannya, karena sudah disediakan sendiri oleh browser yang digunakan. Lain halnya dengan bahasa“Java” yang memerlukan penterjemah khusus untuk menjalankannya di sisi user/klien. Berikut ini satu tabel yang berisi beberapa perbandingan mendasar antara Java dan Javascript.

Javascript	Java
Bahasa yang dijalankan langsung oleh browser	Bahasa yang dijalankan oleh program eksternal (Java Virtual Machine) diintegrasikan ke dalam browser
Didefinisikan dengan tag <script> pada halaman HTML	Didefinisikan dengan tag <applet>
Kode program bersifat terbuka dan dapat dibaca melalui browser	Kode program dapat bersifat tertutup
Tidak memerlukan proses kompilasi menjadi file lainnya	Perlu kompilasi untuk menghasilkan file .class

Javascript adalah bahasa yang “case sensitive” artinya membedakan penamaan variabel dan fungsi yang menggunakan huruf besar dan huruf kecil,

contoh variabel atau fungsi dengan nama TEST berbeda dengan variabel dengan nama test. Dan yang terakhir seperti bahasa Java ataupun C, walaupun hal ini tidaklah wajib, setiap instruksi dapat diakhiri dengan karakter titik koma (;).

Penulisan Kode Javascript

Ada beberapa cara untuk menuliskan kode Javascript di dalam dokumen/halaman HTML, yakni menggunakan tag <script>, menggunakan file eksternal, dan melalui event tertentu.

Menggunakan tag `<SCRIPT>`

Kode javascript dalam HTML dituliskan dengan menggunakan tag <script> dengan format sebagai berikut :

```
<script language="javascript">
......
</script>
```

Atribut language di dalam tag <SCRIPT> menunjukkan jenis bahasa yang digunakan dan versinya, contohnya “Javascript“, “Javascript1.1“,“Javascript1.2“ untuk bahasa Javascript atau bahasa lainnya, contohnya “VBScript“. Jika kita ingin menggunakan beberapa versi Javascript di dalam satu halaman HTML (untuk menyesuaikan dengan kompatibilitas navigator) , maka kita hanya perlu meletakkankode-kode Javascript tersebut (berdasarkan versinya) kedalam beberapa tag <SCRIPT> dengan mencantumkan versi Javascriptnya.

Menggunakan file eksternal

Cara berikutnya adalah menuliskan kode program Javascript dalam suatu file teks dan kemudian file teks yang berisi kode Javascript di panggil dari dalam dokumen HTML (khusus Netscape mulai versi 3 keatas). Kode yang kita sisipkan kedalam dokumen HTML adalah sebagai berikut :

```
<script language="javascript" src="lokasi/file.js"></script>
```

dimana lokasi/file.js adalah adalah lokasi dan nama file yang berisi kode Javascript, jika perintah tambahan SRC tidak disertakan maka tag Script akan mencari kode yang terletak di dalam tag Script.

Melalui event tertentu

Event adalah sebutan dari satu aksi yang dilakukan oleh user, contohnya seperti klik tombol mouse, pembahasan lebih lanjut ada di materi lainnya. Kodenya dapat di tulis sebagai berikut :

```
<tag eventHandler="kode Javascript yang akan dimasukkan">
```

dimana eventHandler adalah nama dari event tersebut.

Dokumentasi Program

Seperti dalam banyak bahasa pemrograman lainnya, sangat dianjurkan untuk menambahkan komentar-komentar di dalam skrip atau kode program yang dibuat untuk keperluan dokumentasi pengembangan. Hal ini berguna antara lain :

a. sebagai pengingat bagian-bagian khusus di dalam skrip tersebut, jika akan merubah sesuatu di dalamnya, mungkin beberapa bulan kemudian sudah tidak ingat lagi dengan detail dan alur dari skrip tersebut.
b. untuk membuat orang yang tidak tahu program yang dibuat jadi mengerti dengan membaca petunjuk-petunjuk yang dibuat melalui komentar-komentar.

Penulisan komentar di Javascript, kita bisa menggunakan cara yang sama dengan aturan yang ada di bahasa C/C++ ataupun Java.

a. Untuk menulis komentar dalam satu baris kita gunakan karakter garis miring ganda (//).
   // semua karakter di belakang
   // tidak akan di eksekusi
b. Untuk menulis komentar yang terdiri dari beberapa baris kita gunakan karakter /* dan */
   /* Semua baris antara 2 tanda tersebut
   tidak akan di eksekusi oleh

kompilator */

Contoh program Javascript

Pada contoh berikut ini adalah contoh skrip Javascript didalam suatu dokumen HTML, disini kita akan membuat satu program untuk menampilkan satu kotak dialog (dijelaskan lebih lanjut di bab lainnya) pada saat kita membuka dokumen HTML.file: halo.html

```
<html>
<head>
<title>contoh program javascript</title>
</head>
<body>
<script language="javascript">
alert("Hallo !");
</script>
</body>
</html>
```

Saat program ini dijalankan melalui browser, maka akan ditampilkan satu dialog berisikan pesan "Hallo !" tanpa tanda kutip.

Selain dengan bentuk lengkap seperti diatas, kita juga dapat menuliskannya seperti berikut dengan hasil yang sama.

file: halo_alternatif.html

```
<script language="javascript">
alert("Hallo !");
</script>
```

Alat Bantu Pengembangan

Dalam mengembangan aplikasi web ada beberapa persiapan yang perlu dilakukan, diantaranya adalah menentukan browser dan editor yang akan digunakan. Saat ini sudah banyak bermunculan varian browser, yang paling umum diantaranya Mozilla Firefox, Google Chrome, Microsoft Internet Explorer, Opera, dan Apple Safari. Hampir semuanya mendukung penggunaan

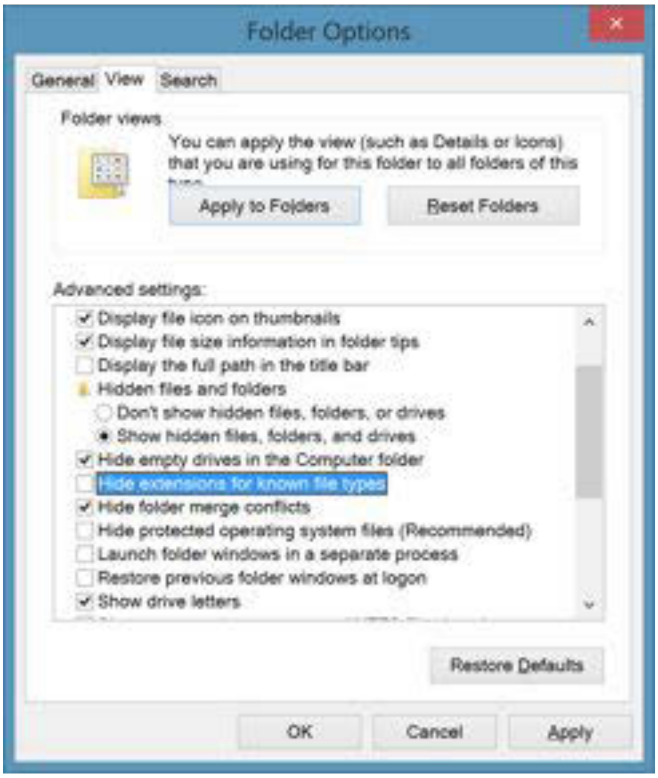
Javascript. Pemilihan browser dapat disesuaikan dengan selera masing-masing. Namun, ada baiknya menginstal lebih dari satu browser untuk mengetahui apakah aplikasi web yang dibuat didukung oleh sebagian besar browser tersebut atau tidak.

Selain browser yang bervariasi, editor pun demikian, sekarang sudah banyak program editor web baik yang tekstual maupun grafis. Diantaranya yang gratis ada Notepad2, Notepad++, Sublime Text, Aptana dan Expression Web 4 atau yang berbayar dengan Adobe Dreamweaver dan Visual Studio. Bagi pengguna Linux dapat menggunakan Gedit, Geany atau Sublime Text. Dalam pengembangan aplikasi web kali ini penulis akan menggunakan Notepad++, aplikasinya dapat didownload pada alamat http://notepad-plus-plus.org.

Panduan Pengembangan di Windows

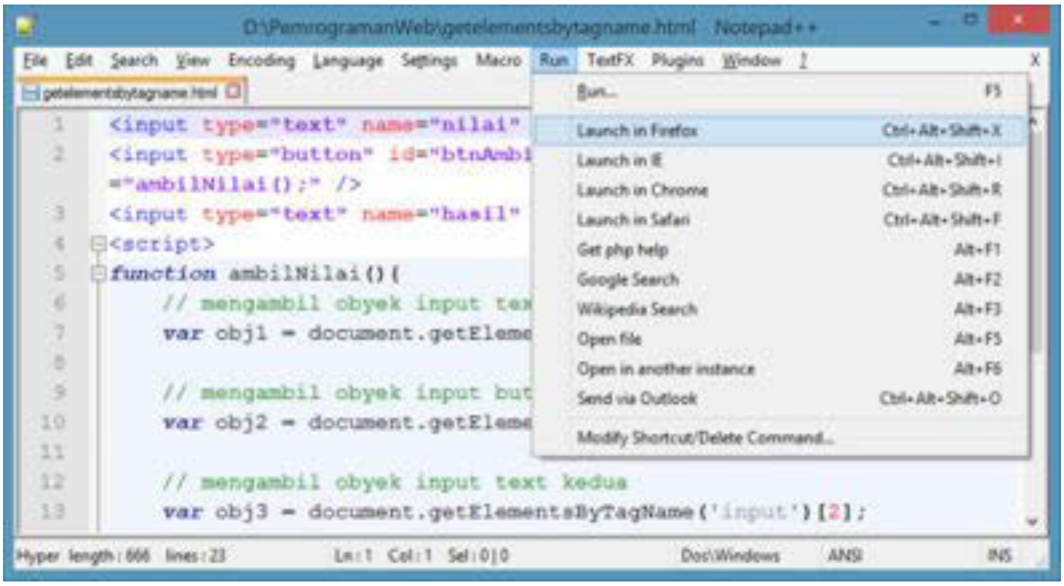
Agar dapat memaksimalkan penggunaan Notepad++ di Windows, dapat mengikuti panduan berikut ini :

a. Pastikkan pada komputer telah terinstall aplikasi browser untuk melihat halaman web yang dihasilkan. Kalau belum bisa coba install browser Mozilla Firefox atau Google Chrome.
b. Tentukan lokasi tempat semua file aplikasi web akan dikembangkan. Misal, di folder D:\PemrogramanWeb.
c. Untuk Windows, pastikkan sistem menampilkan ekstensi file. Jika belum, lakukan langkah berikut:
    i. Untuk Windows 8 keatas: Buka Explorer, pilih menu View kemudian klik Options.
    ii. Untuk Windows 7 kebawah: Buka Explorer, akses menu Tools dan pilih Folder Options.
    iii. Pada window yang tampil pilih tab View, hapus centang pada opsi Hide extensions for known file types.
    iv. Terakhir klik OK.



Gambar 8.1 Tampilan Folder Options pada Windows

d. Setiap kali akan mencoba/membuat program Javascript. Buat terlebih dahulu filenya pada lokasi yang sudah ditentukan diatas. Di Windows dapat dengan klik kanan, trus pilih menu New > Text Document dan ganti namanya sesuai program yang akan dibuat, misalnya: halojavascript.html.
e. Buka file tersebut dengan Notepad++ dan ketikkan programnya.
f. Setelah selesai membuat program, untuk melihat hasilnya klik menu Run > pilih salah satu menu misalnya Launch in Firefox untuk menjalankan file dalam browser Firefox.



Gambar 8.2 Contoh eksekusi program web di Notepad++

Panduan Pengembangan di Linux

Bagi yang ingin mengembangkan aplikasi web di Linux dapat juga mengikuti panduan berikut ini.

a. Editor yang akan dijelaskan disini adalah Geany. Apabila belum diinstall silahkan install dengan menjalankan perintah berikut pada terminal.

   Ubuntu:
   sudo apt-get install geany

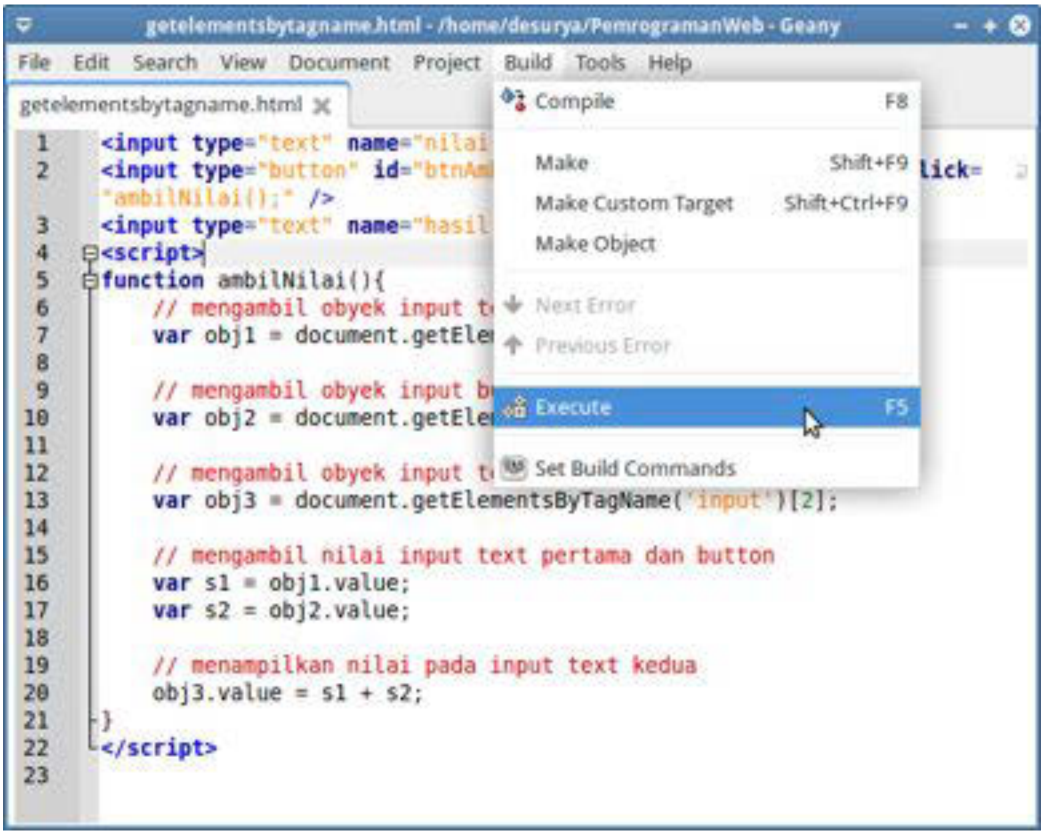
   OpenSuSE:
   sudo zypper install geany

b. Tentukan lokasi pengembangan aplikasi, misalnya di folder /home/namauser/PemrogramanWeb.

c. Setiap kali akan mencoba/membuat program Javascript. Buat terlebih dahulu filenya pada lokasi yang sudah ditentukan diatas. Di OpenSuSE dapat dengan klik kanan, trus pilih menu Create New > Text File dan ganti namanya sesuai program yang akan dibuat, misalnya: halojavascript.html. Untuk Ubuntu dapat menyesuaikan.

d. Buka file tersebut dengan geany dan ketikkan programnya.

e. Setelah selesai membuat program, untuk melihat hasilnya klik menu Build > pilih Execute untuk menjalankan file dalam browser utama yang ada.



Gambar 8.3 Contoh eksekusi program web di editor Geany

c. Rangkuman.

Dari paparan materi pembahasan kegiatan belajar ini dibuat rangkuman materi sebagai berikut :

- Javascript merupakan bahasa yang dapat digunakan untuk mengolah halaman web.
- Program javasacript dapat ditempatkan bersamaan dengan file HTML ataupun secara terpisah dalam file tersendiri.
- Javascript berbeda dengan Java yang dalam implementasinya di HTML menggunakan tag yang lain dari Javascript.
- Komentar diperlukan dalam program untuk membantu dalam pengembangan aplikasi selanjutnya.



d. Tugas.

Buatlah timeline sejarah perkembangan Javascript dalam bentuk dokumen HTML dengan tampilan sebagai berikut.

Timeline Sejarah Javascript

Tahun	Perkembangan
…	…
…	…

e. Tes Formatif.

1. Jelaskan perbedaan antara Java dengan Javascript?
2. Apa peranan browser dalam penerapan Javascript pada suatu dokumen HTML?
3. Mengapa komentar diperlukan dalam pengembangan program?
4. Bagaimana menuliskan kode program javascript pada suatu dokumen HTML?
5. Mungkinkah mengembangkan aplikasi javascript untuk aplikasi berbasis Desktop bukan Web saat ini? Temukan dan jelaskan.

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

9. Kegiatan Belajar 9 :Memahami Dasar Pemrograman Web Klien

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 9 ini siswa diharapkan dapat :

1) Mengetahui dasar-dasar pemrograman web klien
2) Memahami pembuatan kode program dasar web klien

b. Uraian Materi.

VARIABEL

Variable adalah suatu obyek yang berisi data data, yang mana dapat dimodifikasi selama eksekusi program. Di Javascript terdapat beberapa kriteria untuk penamaan variabel, seperti ditunjukkan berikut ini.

- Nama variabel harus dimulai oleh satu huruf (huruf besar maupun huruf kecil) atau satu karakter _ (garis bawah).
- Nama variabel bisa terdiri dari kombinasi huruf, angka atau karakter _ dan $ (spasi kosong tidak diperbolehkan).
- Nama variabel tidak boleh memakai nama-nama yang termasuk ke dalam kata kunci bahasa Javascript.

Berikut ini merupakan daftar kata kunci yang ada di Javascript.

abstract	else	int	synchronized
boolean	enum	interface	this
break	export	let	throw
byte	extends	long	throws
case	final	native	transient
catch	finally	new	try
char	float	package	typeof
class	for	private	var
const	function	protected	void
continue	goto	public	volatile
debugger	if	return	while
default	implements	short	with
delete	import	static	yield
do	in	super	
double	instanceof	switch	

Sumber: kompilasi https://developer.mozilla.org dan http://msdn.microsoft.com.

Berikut ini adalah contoh pemberian nama variabel yang benar dan tidak benar:

Nama variabel yang benar

- umur
- namaLengkap
- luas_lingkaran
- jari2
- _7an
- $biaya

Nama variabel yang tidak benar

- Nama Lengkap → Ada spasi kosong
- 7ruangan → Dimulai dgn angka
- email@website.com → Karakter @
- top-level-domain → Karakter -
- function → Kata kunci Javascript

Di Javascript kita menggunakan sistem case sensitive yang artinya membedakan nama variabel dengan huruf besar dan huruf kecil, oleh karena itu biasakanlah memberikan nama variabel dengan aturan yang sama.

Ada beberapa teknik pemberian nama yang dapat digunakan, diantaranya adalah

- Camel Case, dengan teknik untuk nama variabel yang lebih dari satu kata akan digabungkan tanpa garisbawah dan setiap huruf pertama dari tiap kata dituliskan dalam huruf kapital. Sedangkan untuk huruf pertama dari kata pertama dapat juga ditulisakan menggunakan huruf kecil. Contoh: AsalSekolah, SuhuUdara, luasPersegiPanjang, namaLengkap, dan sebagainya.
- Notasi Hungaria, disini diawal dari nama variabel akan ditambahkan tipe data yang ditampungnya, seterusnya dapat menggunakan notasi camel case. Contoh: stringNamaLengkap (atau lebih pendek, strNamaLengkap,

string menunjukkan variabel ini berisi data alfanumerik), intUmur (int menunjukkan variabel ini menyimpan bilangan bulat), dan sebagainya.

- Selain dengan camel case, variabel dengan nama lebih dari satu kata dapat juga dibuat dengan menambahkan garisbawah sebagai penghubung antar katanya. Contoh: asal_sekolah, luas_persegi, nama_lengkap, dan sebagainya.

PEMBUATAN VARIABEL

Penulisan variabel Javascript sangatlah fleksibel, dan tidaklah terlalu rumit dan ketat, sehingga akan jarang menerima pesan error karena variabel yang tidak sesuai.

Ada dua istilah yang biasa digunakan dalam pembuatan variabel, yakni deklarasi dan inisialisasi. Deklarasi digunakan untuk menunjukkan proses permintaan alokasi memori oleh Javascript ke sistem operasi. Sedangkan inisialisasi digunakan untuk memberikan nilai awal bagi variabel tersebut.

Format deklarasi (pembuatan) variabel di Javascript adalah sebagai berikut :

```
var nama_variabel_nya ;
```

atau, dengan tambahan inisialisasi :

```
var nama_variabel = nilai ;
```

atau untuk deklarasi variabel lebih dari satu dalam satu baris perintah :

```
var nama_variabel1, nama_variabel3 = nilai_var_3, …, nama_variabelN ;
```

contoh :

```
var umur;
var panjang = 23;
```

```
var alas, tinggi;
<script language="javascript">
var nilai;
var nilai2 = 3;
nilai = 2;
document.write(nilai * nilai2);
</script>
```

TIPE DATA

Di javascript data yang dapat disimpan dalam suatu variabel dikelompokkan menjadi tiga jenis, yaitu :

- Bilangan/numerik, bulat atau desimal, yang kita sebut sebagai integer atau float
- Kata/kalimat (kumpulan huruf/karakter) : kita sebut string
- Boolean : suatu variabel yang mempunyai dua nilai dan berfungsi untuk memeriksa suatu kondisi :
- true : jika kondisinya benar
- false : jika kondisinya salah

INTEGER (BILANGAN BULAT)

bilangan bulat dapat ditampilan dalam beberapa basis berikut ini :

- Basis desimal, integer di tuliskan dalam urutan unit bilangan (dari 0 sampai dengan 9), permulaan bilangan tidak boleh dimulai oleh angka 0.
- Basis heksadesimal, dituliskan dalam urutan unit bilangan dari 0 sampai dengan 9 atau urutan huruf dari A sampai dengan F (atau a sampai dengan f), permulaan bilangan dimulai oleh 0x atau 0X.
- Basis oktal, dituliskan dalam urutan unit angka dari 0 sampai dengan 7, permulaan bilangan dimulai dengan angka 0.

FLOAT (BILANGAN DESIMAL)

Bilangan desimal biasa disebut juga sebagai bilangan pecahan atau bilangan yang dapatdituliskan menggunakan tanda koma. Bilangan ini juga bisa di tuliskan denganbeberapa cara berikut :

- Bilangan bulat desimal : 895
- Bilangan dengan tanda koma : 895,12
- Bilangan pembagian : 27/11
- Bilangan eksponensial : bilangan dengan tanda koma, kemudian diikuti oleh huruf e(atau E), kemudian diikuti oleh bilangan bulat yang artinya pangkat dari bilangan 10 (+ atau -, pangkat postitif atau negatif), contoh :

```
var a = 2.75e-2 ;
var b = 35.8E+10 ;
var c = .25e-2 ;
```

STRING

String adalah kumpulan dari karakter/huruf, nilai variabel string selalu diapit dengan tanda (') atau ("), kedua tanda tersebut harus digunakan secara berpasangan dan tidak bisa digunakan secara sendiri-sendiri atau bersilangan. Berikut ini adalah beberapa cara untuk mendeklarasikan variabel string :

```
var a = "Hallo";
var b = 'Sampai Ketemu Lagi !';
```

Contoh yang tidak sesuai, berikut ini :

```
var a = ‘Hallo”;
var v = “Sampai Ketemu Lagi !';
```

Ada beberapa karakter spesial yang bisa kita gunakan untuk mensimulasikan bagian dari karakter yang tidak terlihat (non visual) dan juga untuk menghindarkan kemungkinan browser "mengalami kebingungan" dalam membedakan antara string dan skripnya sendiri, karakter spesial ini menggunakan simbol antislash (\), beberapa contoh karakter spesial tersebut:

- \n: kembali ke baris awal
- \r: menekan tombol ENTER
- \t: tab
- \": tanda petik ganda
- \': tanda petik tunggal
- \\: karakter antislash

Contoh, misalnya kita ingin menyimpan kalimat berikut ini :

```
Ada apa di dalam "c:\windows\"
```

Kita harus menuliskannya dalam bentuk berikut ini di dalam Javascript :

```
var Judul = "Ada apa di dalam \"c:\\windows\\\"";
```

Atau bisa juga dengan cara berikut ini (menggunakan tanda petik tunggal) :

```
Judul = 'Ada apa di dalam "c:\\windows\\"';
```

Memanipulasi variabel String, dapat dilakukan dengan menggunakan metode-metode yang ada pada variabel string tersebut. Metode-metodenya dapat dilihat pada materi tentang Fungsi.

BOOLEAN

boolean adalah satu variabel khusus yang berguna untuk mengevaluasi suatu kondisi tertentu, oleh karenanya boolean mempunyai dua nilai :

- True : diwakili oleh nilai 1
- False : diwakili oleh nilai 0

KONVERSI JENIS VARIABEL

Meskipun Javascript memungkinkan pengaturan perubahan jenis variabel secara transparan,kadang kadang kita perlu juga untuk melakukan konversi jenis variabel secara paksa. parseInt() dan parseFloat() merupakan dua fungsi yang dapat digunakan untuk konversi ini.

Fungsi parseInt()

Fungsi ini mungkinkan merubah satu variabel yang dilewatkan dengan parameter tertentu (bisadalam bentuk string ataupun dalam bentuk bilangan dalam basis yang disebutkan di parameterkedua) menjadi bilangan bulat. Sintaksnya adalah sebagai berikut :

```
parseInt(string[, basis]);
```

Agar supaya fungsi parseInt() mengembalikan nilai bilangan bulat, maka parameter yangdilewatkan harus dimulai dengan karakter bilangan [0-9], prefiks

hexadesimal 0x, dan/ataukarakter +,-,e,dan E. Selain daripada itu maka fungsi parseInt() akan mengembalikan nilai NaN(Not a Number). Jika karakter berikutnya tidak valid, maka akan diabaikan oleh fungsiparseInt(), dan akan ditampilkan terpotong jika di bagian depan karakter valid dan bagianbelakang karakter tidak valid.

Berikut ini salah satu contoh penggunaan fungsi parseInt() :

```
var a = "123";
var b = "456";
document.write(a+b + "<BR>"); // hasil 123456
document.write(parseInt(a)+parseInt(b)  +  "<BR>");  //  hasil
579
```

Table 1. Contoh penggunaan fungsi parseInt()

Contoh	Hasil
parseInt("128.34");	128
parseInt("12.3E-6");	12
parseInt("12E+6");	12
parseInt("Halo");	NaN
parseInt("24Halo38");	24
parseInt("Halo3824");	NaN
parseInt("AF8BEF");	NaN
parseInt("0284");	284
parseInt("0284",8);	2
parseInt("AF8BEF",16);	11504623
parseInt("AB882F",16);	11241519
parseInt("0xAB882F");	11241519
parseInt("0xAB882F",16);	11241519
parseInt("00100110");	100110
parseInt("00100110",2);	38
parseInt("00100110",8);	32840
parseInt("00100110",10);	100110
parseInt("00100110",16);	1048848

Fungsi parseFloat()

Adalah satu fungsi dasar dari Javascript yang memungkinkan merubah variabel yang dilewatkandengan parameter tertentu menjadi bilangan desimal, Sintaks dari fungsi parseFloat() adalah sebagai berikut :

```
parseFloat(string);
```

Table 2. Contoh penggunaan fungsi parseFloat()

Contoh	Hasil
parseFloat("128.34");	128.34
parseFloat("128,34");	128
parseFloat("12.3E-6");	0.0000123
parseFloat("Halo");	NaN
parseFloat("24.568Halo38");	24.568
parseFloat("Halo38.24");	NaN
parseFloat("AF8BEF");	NaN
parseFloat("0284");	284
parseFloat("0xAB882F");	11241519

OPERATOR

Operator adalah simbol yang dapat digunakan untuk memanipulasi nilai suatu variabel, dengan kata lain melakukan operasi-operasi, mengevaluasi, dan lain-lainnya. Ada beberapa jenis operator :

OPERATOR PENGHITUNGAN

Operator penghitungan digunakan untuk memodifikasi nilai dari variabel secara matematika.

Table 3. contoh penggunaan fungsi Operator Perhitungan

Operator	Fungsi	Contoh	Hasil y (dengan x bernilai 7)
+	Menjumlahkan dua nilai	y = x + 3	10

-	Mengurangi nilai suatu variabel dengan nilai lain	y = x - 3	4
*	Mengalikan dua bilangan	y = x * 3	21
/	Membangi nilai suatu variabel dengan nilai lainnya	y = x / 3	2.333
=	Memberikan suatu nilai terhadap variabel dikiri operator ini	y = 7	7
%	Mencari sisa hasil bagi daru dua nilai	y = x % 3	1
+=	Menjumlahkan nilai variabel dengan nilai tertentu dan hasilnya disimpan di variabel yang sama	x += 3	10
-=, *=, /=	Memiliki prinsip yang sama dengan operator +=	x -= 3 x *= 3 x /= 3	4 21 2.333
++, --	Digunakan untuk menambahkan atau mengurangi 1 nilai suatu variable	x++ x--	8 6

OPERATOR LOGIKA (BOOLEAN)

Operator jenis ini memungkinkan kita untuk memverifikasi apakah beberapa kondisi sudah benar

Table 4.contoh penggunaan fungsi Operator Logika

Operator	Fungsi	Contoh	Hasil (dengan x = 7)
==	Membandingkan nilai di kiri dan kanan operator ini	x == 7	true, jika x bernilai 7 false, jika x bernilai bukan 7
===	Sama dengan ==, tambahannya untuk menyamakan tipe datanya juga	x === "7"	true, jika x juga adalah string bukan integer dan nilainya 7.

!=	Mengetahui apakah dua nilai tidak sama	x != 7	true, jika nilai x bukan 7 false, jika nilai x adalah 7
!==	Mencari ketidaksamaan nilai dan tipe datanya	x !== “7”	true, jika nilai bukan 7 dan x bukan string false, jika nilai x adalah 7 dan x adalah string
<	Membandingkan apakah bilangan yang dikiri lebih kecil dari yang dikanan operator	x < 7	true, jika x lebih kecil dari 7 false, jika x lebih besar atau sama dengan 7
<=	Membandingkan apakah bilangan yang dikiri lebih kecil atau sama dengan yang dikanan	x <= 7	true, jika x lebih kecil atau sama dengan 7 false, jika x lebih besar dari 7
>, >=	Prinsipnya kebalikan dari operator < dan <=		
||	Logika OR untuk dua kondisi yang diberikan	x > 7 || x == 7	true, jika nilai x lebih besar dari 7 ATAU x sama dengan 7 false, jika nilai x tidak lebih besar dari 7 DAN nilai x tidak sama dengan 7
&&	Logika AND	x > 7 && x < 10	true, jika nilai x lebih besar dari 7 DAN x lebih kecil dari 10 false, jika nilai x tidak lebih besar dari 7 ATAU x tidak lebih kecil dari 10

!	Logika NOT, membalikan nilai boolean suatu variable	!(x == 7)	true, jika nilai x tidak sama dengan 7 false, jika nilai x adalah 7

OPERATOR STRING

Operator + pada data berjenis string digunakan untuk menggabungkan dua variabel/nilaistring. Perlu dicatat juga bahwa var tes ='a'+'b' adalah sama dengan var tes ='ab'.

Contoh :

```
var tes ='a';
var tes2 = tes +'b'; //hasilnya: tes2 akan bernilai string
"ab"
```

PRIORITAS

Pada saat akan melakukan operasi dengan banyak operator dalam satu waktu, browser harus tahu dengan urutan mana operasi dilakukan berdasarkan prioritas dari operator. Prioritas ini menunjukkan urutan suatu operator dieksekusi oleh browser. Berikut ini adalah tabel tingkat prioritas dari seluruh operator mulai dari yang tertinggi.

Table 5. contoh penggunaan fungsi Urutan eksekusi operator

Urutan	Operator
1	. [] ()
2	++ -- - ~ ! delete new typeof void
3	* / %
4	+ - +
5	<<>>>>>
6	<<= >>= instanceof
7	== != === !==

<table>
<tr><td>8</td><td>&</td></tr>
<tr><td>9</td><td>^</td></tr>
<tr><td>10</td><td>|</td></tr>
<tr><td>11</td><td>&&</td></tr>
<tr><td>12</td><td>||</td></tr>
<tr><td>13</td><td>?:</td></tr>
<tr><td>14</td><td>= += -= *= /= %= <<= >>= >>>= &= ^= |=</td></tr>
<tr><td>15</td><td>,</td></tr>
</table>

Contoh :

```
var hasil = 78 * 9 + 3;
document.write(hasil);// hasilnya: 702
document.write("<br>");
hasil = 78 * (9 + 3);
document.write(hasil); // hasilnya: 936
```

c. Rangkuman.

Dari kegiatan belajar ini dapat dibuat ringkasan materi sebagai berikut :

- Dasar-dasar dalam pemrograman javascript meliputi variabel, tipe data, dan operator.
- Setiap operator yang ada memiliki prioritas dalam eksekusinya apabila ditempatkan bersama dalam satu operasi.

d. Tugas.

1. Buatlah program javascript untuk menghitung luas lingkaran jika diketahui jari-jarinya adalah 2.3?
2. Diketahui variabel B bernilai 2.3, gunakan program javascript untuk memisahkan/mengambil bagian berkoma dari bilangan diatas sehingga misal, variabel C akan bernilai 0.3?
3. Berapakah nilai rata-rata dari kumpulan nilai berikut:

   2.3 3.4 4.5 6.7 7.8 8.9 9.10 10.1

   Gunakan program javascript untuk menghitung nilai rata-ratanya?

e. Tes Formatif.

1. Apakah yang dimaksud dengan variabel?
2. Apa saja data yang dapat diolah oleh Javascript?
3. Apa perbedaan antara operator + pada operasi string dan numerik?
4. Apa perbedaan antara operator = di pemrograman dan matematika?
5. Mengapa dalam prioritas operator tanda () mendapat prioritas tertinggi?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

10. Kegiatan Belajar 10 : Memahami Penggunaan Array

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 10 ini siswa diharapkan dapat :

1) Mengetahui keberadaan nilai jamak dalam suatu program
2) Memahami penggunaan array dalam program

b. Uraian Materi.

Variabel-variabel di JavaScript hanya bisa menyimpan satu data pada suatu saat. Adanya kecenderungan yang besar untuk memanipulasi sejumlah data dalam satu variabel membuat konsep variabel tersebut menjadi tidak cukup. Sebagai solusinya, JavaScript menghadirkan struktur data yang memungkinkan menyimpan himpunan/kelompok data dalam satu variabel khusus yang disebut Array.

Array, adalah satu variabel yang dapat berisi banyak data yang independent. Data yang disimpan diberikan indeks tertentu atau menggunakan nomer urut mulai dari 0.Indeks ini nantinya dapat digunakan untuk mengakses data yang disimpan dalam array tersebut.

JENIS-JENIS ARRAY

Array Multidimensi

Array yang hanya berisikan nilai-nilai tunggal disebut sebagai array monodimensi atau unidimensi. Berikut merupakan ilustrasi dari array monodimensi tersebut.

Indeks	0	1	2	3
Data	Nilai1	Nilai2	Nilai3	Nilai4

Perhatian :

Indeks ini akan diberikan secara otomatis oleh interpreter javascript, dimulai dari elemen/data pertama dengan indeks 0. Dalam suatu array dengan n elemen, maka elemen ke-n akan mempunyai indeks n-1.

Jika elemen suatu array berisikan array lainnya, maka array tersebut termasuk array multidimensi. Berikut ilustrasinya.

<table>
<tr><th>Indeks</th><th colspan="2">0</th><th>1</th><th>2</th><th colspan="3">3</th></tr>
<tr><td rowspan="2">Data</td><td>0</td><td>1</td><td rowspan="2">Nilai2</td><td rowspan="2">Nilai3</td><td>0</td><td>1</td><td>2</td></tr>
<tr><td>Nilai1</td><td>Nilai2</td><td>Nilai1</td><td>Nilai2</td><td>Nilai3</td></tr>
</table>

Array Asosiatif

Indek dalam array selain ditentukan secara otomatis oleh sistem dan selalu dimulai dari 0, juga dapat dimodifikasi dengan menggunakan kata (string) atau nilai tertentu. Tabel dengan indeks termodifikasi ini kita kenal dengan nama array asosiatif. Berikut adalah ilustrasinya.

Indeks	“USD”	“IDR”	“JPY”
Data	11918	1	114.29

PEMBUATAN ARRAY

Terdapat beberapa cara yang dapat digunakan untuk membuat array di Javascript.

```
var TabelKu = ["data 1", "data 2", "data 3", "data 4"];
var TabelKu = new Array("data 1", "data 2", "data 3", "data 4");
var TabelKu = new Array();
```

Dua cara yang pertama menunjukkan bahwa array dibuat dengan langsung memberikan nilainya, sedangkan pada cara yang terakhir array dibuat tanpa data.

PENGAKSESAN DATA DALAM ARRAY

Pengaksesan data atau elemen di dalam tabel dapat dilakukan dengan menuliskan nama tabel diikuti tanda kurung siku yang berisi indeks dari elemen.

```
var arrayHewan = ["Buaya", "Harimau", "Gajah", "Singa",
"Jerapah", "Zebra"];
document.write( "elemen ke 4 dari array adalah " +
arrayHewan[3] );
//hasil: "elemen ke 4 dari array adalah Singa"
```

MENYIMPAN DATA PADA ARRAY

Penyimpanan data dalam array dilakukan dengan menuliskan nama array, di ikuti dengan nama indeks dalam kurung, dan memasukkan nilai datanya, seperti contoh berikut.

```
arrayTes[0] = "Hallo";
arrayTes["Andry"] = 10;
arrayTes["Sandra"] = 47;
```

Apabila indeks yang digunakan untuk menunjuk elemen dalam array tersebut sudah memiliki data sebelumnya, maka data tersebut akan hilang dan digantikan dengan data baru yang diberikan.

c. Rangkuman.

Dari kegiatan belajar ini dapat dibuat rangkuman materi sebagai berikut :

- Array, konsep yang dikembangkan untuk memungkinkan suatu variabel menampung lebih dari satu data dalam satu waktu.
- Setiap data dalam array disimpan dan diberikan index numerik mulai dari 0 untuk elemen pertama dan index n – 1 untuk elemen ke-n.
- Seperti layaknya variabel, pada saat menyimpan data pada index tertentu dari array, apabila sebelumnya ada data yang disimpan disana, data lama tersebut akan dihapus dan digantikan dengan data yang baru.

d. Tugas.

1. Gunakan konsep array diatas untuk menampung data bulan dan zodiak dalam array masing-masing?
2. Diketahui himpunan namaHari = {“Senin”, “Selasa”, “Rabu”, “Kamis”, “Jumat”, “Sabtu”, “Minggu”} dan himpunan bilanganBulat = {11, 12, 13,

14, 15, 16, 17}, memiliki relasi satu-satu, dimana setiap satu bilangan bulat berpasangan tepat dengan satu namaHari, seperti gambar dibawah. Buatlah program javascript untuk menunjukkan relasi ini, sehingga apabila bilangan bulat yang dipilih adalah 12, maka hari yang ditampilkan adalah “Selasa”.

Dibawah ini adalah contoh potongan kode yang mampu menghasilkan program diatas. Pada baris terakhir document.write() akan menampilkan nilai dari variabel elemenHari yakni “Selasa”.

```
…
var  elemenBilBulat = 12;
var elemenHari = “”;
…
document.write(elemenHari);
```

e. Tes Formatif.

1. Apakah yang dimaksud dengan array?
2. Jelaskan jenis array yang dapat dihasilkan di Javascript?
3. Sebutkan mekanisme operasi apa saja yang dapat dilakukan terhadap suatu array?
4. Apa yang terjadi pada suatu elemen dari array apabila ada data baru yang disimpan disana?
5. Mungkinkah membuat suatu array di Javascript dengan index elemen pertama dimulai dari 1?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

11. Kegiatan Belajar 11 : Memahami Penerapan Struktur Kontrol Percabangan

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 11 ini siswa diharapkan dapat :

1) Mengetahui format percabangan dalam program
2) Memahami penerapan percabangan dalam program

b. Uraian Materi.

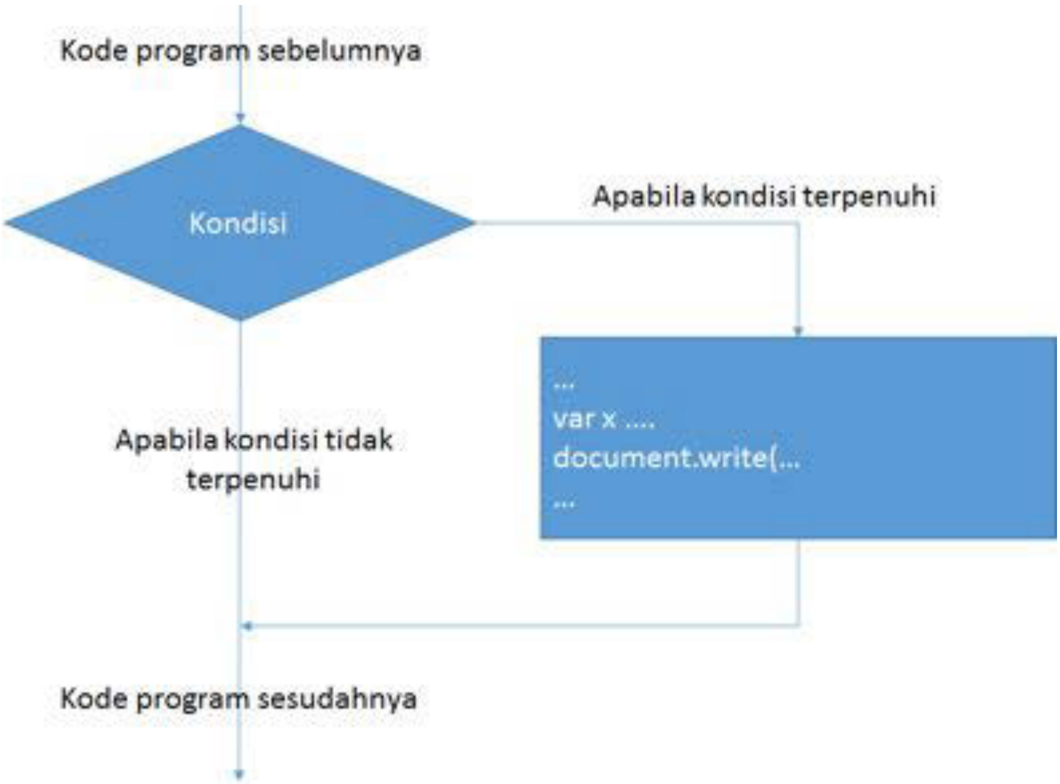
Struktur Kontrol

Struktur kontrol merupakan konsep pemrograman dikembangkan sedemikian rupa sehingga dapat digunakan untuk menghasilkan keluaran tertentu sesuai nilai-nilai dari variabel-variabel yang ada. Struktur kontrol dapat juga diartikan sebagai rangkaian perintah yang dapat digunakan untuk mengendalikan alur dari program. Secara alur, program memiliki tiga jenis alur, yakni berurutan, berulang, dan bercabang.

Struktur berurutan dapat digambarkan sebagai kumpulan program yang dijalankan mengikuti alur tertentu dapat secara vertikal dari atas ke bawah atau mengikuti alur yang sudah ditentukan tanpa adanya kondisi-kondisi khusus yang memungkinkan adanya pilihan alur. Struktur bercabang ataupun berulang berguna untuk menjelaskan alur program yang memiliki kondisi-kondisi khusus ini. Struktur percabangan dapat digunakan untuk mengeksekusi bagian tertentu dari program apabila suatu kondisi terpenuhi. Demikian juga dengan perulangan, dapat digunakan untuk menjalankan beberapa kali suatu bagian tertentu dari program dengan terpenuhinya suatu kondisi.

Struktur Percabangan

Konsep percabangan dalam pemrograman mirip seperti konsep percabangan di jalan raya, yang penentuan arahnya ditentukan berdasarkan tujuan dari user. Arah disini menunjukkan bagian program yang akan dieksekusi, sedangkan tujuan merupakan data masukan yang digunakan percabangan untuk mengambil keputusan. Secara umum percabangan digambarkan sebagai berikut.



Gambar 11.1 Gambaran umum struktur percabangan

Dari gambar diatas dapat dilihat, bahwa apabila suatu kondisi terpenuhi maka blok program dalam kondisi tersebut akan dijalankan. Jika dituliskan dalam format kode program dinyatakan seperti berikut ini :

```
if ( kondisi ) {
   kode program yang akan dijalankan jika kondisi terpenuhi;
}
```

Terpenuhinya suau kondisi ini secara pemrograman berarti bahwa kondisi tersebut memiliki nilai boolean true. Blok program yang akan dijalankan ditempatkan di dalam kurung kurawal. Berikut ini merupakan contoh program dengan kondisi yang akan menghasilkan nilai true atau false sesuai input umur yang diberikan.

```
var umur;
umur = prompt("masukkan umur anda?");
```

```
umur = parseInt(umur);
if ( umur < 20 ) {
    alert ( "Kamu belum dapat mengikuti pemilu" );
}
```

Kondisi pada program diatas menunjukkan bahwa apabila input dari user bernilai lebih kecil dari 20, maka kode dalam blok if tersebut akan dijalankan, yakni menampilkan pesan "Kamu belum dapat mengikuti pemilu" tanpa tanda kutip.

Pembuatan kondisi dilakukan dengan menggunakan operator logika. Daftar operator logika dapat dilihat kembali pada tabel operator logika di materi tentang Dasar Pemrograman Web Klien. Pembuatan kondisi lebih dari satu dapat dilakukan dengan menggunakan kombinasi operator logika yang ada, misalnya kita ingin menampilkan pesan "Kamu mungkin adalah siswa SMA/K" apabila umur dari siswa tersebut lebih besar dari 15 dan lebih kecil dari 21 atau dapat juga dikatakan bahwa umurnya diantara 16-20, programnya dituliskan seperti berikut.

```
var umur;
umur = prompt("masukkan umur anda?");
umur = parseInt(umur);
if ( umur > 15 && umur < 21 ) {
    alert ( "Kamu mungkin adalah siswa SMA/K" );
}
```

Contoh program lainnya.

```
var olahraga;
olahraga = prompt("masukkan olahraga yang kamu sukai?");
if ( olahraga == "terjun payung" || olahraga == "sepeda
gunung" || olahraga == "panjat tebing" )
{
    alert("Kamu menyukai olahraga ekstrim");
}
```

Berikut ini merupakan format lainnya yang dapat digunakan untuk menerima kondisi yang tidak sesuai.

```
if (kondisi) {
    kode program;
}
else {
    kode program jika kondisi tidak terpenuhi;
}
```

Untuk kondisi yang lebih dari satu dapat juga dibuat menggunakan format berikut.

```
if (kondisi 1) {
    kode program jika kondisi 1 terpenuhi;
}
else if (kondisi 2){
    program jika kondisi 2 terpenuhi;
}
else if (kondisi n){
    kode program jika kondisi n terpenuhi;
}
else {
    kode program jika kondisi tidak terpenuhi;
}
```

Switch

Switch merupakan bentuk lain dari percabangan. Konsep ini berguna apabila kondisi yang akan ditentukan memiliki nilai tertentu. Format dari switch adalah sebagai berikut.

```
switch(variabel) {
    case nilai1:
      kode program jika nilai variabel sama dengan nilai1;
      break;
    case nilaiN:
```

```
    kode program jika nilai variabel sama dengan nilaiN;
    break;
  default:
    kode program jika nilai tidak sesuai;
    break;
}
```

Contoh kasus untuk format ini adalah pada pemilihan menu layanan isi ulang, seperti contoh berikut :

```
var pilihan = prompt("Pilihan untuk cek pulsa\n1. Informasi
pulsa anda\n2.
Informasi masa aktif dan masa tenggang anda");
pilihan = parseInt(pilihan);
switch(pilihan) {
    case 1:
        alert("Jumlah pulsa anda adalah sembilan belas
ribu
        rupiah");
        break;
    case 2:
        alert("Masa aktif kartu anda hingga 7 Agustus
2019");
        break;
    default:
        alert("Pilihlah sesuai nomor yang ada");
        break;
}
```

Pada contoh program diatas apabila user memilih 1, maka akan ada pesan "Jumlah pulsa anda adalah sembilan belas ribu rupiah" yang ditampilkan. Apabila pilihannya tidak ada dalam blok switch-case tersebut maka bagian default yang akan dijalankan.

Pernyataan break pada program diatas digunakan untuk keluar dari blok case atau default yang terpilih. Apabila tidak terdapat break maka blok case dan/atau default dibawahnya akan turut dieksekusi.

c. Rangkuman.

Dari kegiatan belajar diatas dapat dibuat rangkuman materi sebagai berikut :

- Struktur kontrol dikembangkan untuk membuat program mampu menghasilkan output yang beragam tergantung input yang diberikan oleh user.
- Percabangan merupakan salah satu bentuk struktur kontrol yang memungkinkan untuk menjalankan bagian tertentu dari program apabila kondisinya terpenuhi.
- Selain menggunakan if-else, bentuk implementasi lain dari percabangan adalah dengan menggunakan switch. Switch dapat digunakan pada kasus dimana input yang dikondisikan memiliki nilai tertentu.

d. Tugas.

1. Buatlah program untuk dapat mengetahui index berat badan seseorang beserta pengelompokkannya?
2. Buatlah program untuk mengetahui zodiak dari seseorang berdasarkan input tanggal lahirnya?

e. Tes Formatif.

1. Apa yang dimaksud dengan struktur kontrol dalam pemrograman?
2. Apakah perbedaan antara percabangan if-else dengan switch?
3. Mungkinkah menggunakan variabel string sebagai variabel dalam kondisi blok switch? Jelaskan.
4. Mungkinkah ada blok switch dalam blok if-else atau sebaliknya dalam program? Tunjukkan dengan menggunakan contoh.

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

12. Kegiatan Belajar 12 : Memahami Penerapan Struktur Kontrol Perulangan

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 12 ini siswa diharapkan dapat :

1) Mengetahui format perulangan dalam program
2) Memahami penerapan perulangan dalam program

b. Uraian Materi.

Struktur Kontrol Perulangan

Perulangan merupakan struktur yang memungkinkan untuk menjalankan kode program yang sama beberapa kali dengan menggunakan suatu kondisi tertentu. Perulangan dalam javascript memiliki tiga bentuk, yakni perulangan for, while dan do-while.

Perulangan for

Perulangan ini memiliki bentuk sebagai berikut :

```
for ( inisialisasi; kondisi; peubah ) {
     baris program 1;
     baris program 2;
     …
}
```

Perulangan while

Perulangan ini memiliki bentuk sebagai berikut :

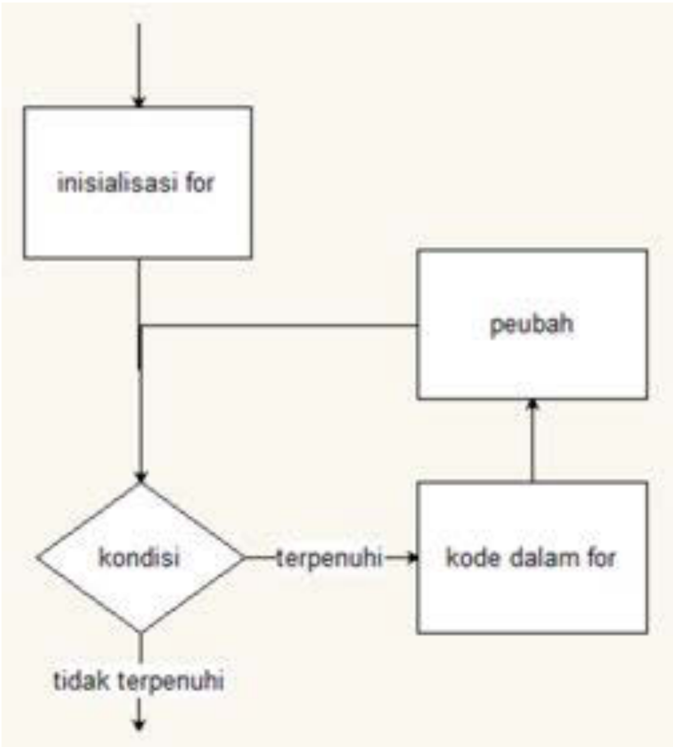
```
while ( kondisi ) {
     kode program yang ingin dijalankan berkali-kali;
}
```

Perulangan do-while

Perulangan ini memiliki bentuk sebagai berikut :

```
do {
    kode program yang ingin dijalankan berkali-kali;
} while( kondisi );
```

Dari ketiga bentuk perulangan diatas hanya for yang memerlukan pernyataan inisialisasi dan peubah. Inisialisasi disini digunakan sebagai bagian penentuan variabel yang akan digunakan untuk mengendalikan jumlah perulangan yang akan dilakukan. Peubah adalah operasi aritmatika yang dapat dilakukan untuk mengubah nilai variabel penentu tersebut. Sedangkan kondisi disini digunakan untuk menentukan kapan suatu blok perulangan dijalankan berdasarkan nilai dari variabel penentu tersebut. Cara kerja dari perulangan for tersebut dapat digambarkan dengan menggunakan diagram alur berikut.



Gambar 12.1 Alur kerja program untuk strutur percabangan

Secara umum untuk setiap kasus perulangan dapat diterapkan dengan menggunakan ketiga bentuk tersebut. Contohnya, kita dapat membuat perulangan untuk menampilkan pesan "Halo javascript" hingga 3 kali dengan menggunakan ketiga perulangan tersebut.

```
for (var i = 0; i < 3; i++) {
    document.write("Halo javascript 1<br>");
}

var j = 0;
while ( j < 3 ) {
    document.write("Halo javascript 2<br>");
    j +=  1;
}

var k = 0;
do {
    document.write("Halo javascript 3<br>");
    k++;
} while ( k < 3 );
```

c. Rangkuman.

Dari paparan kegiatan penerapan struktur kontrol perulangan dapat dibuat ringkasan materi sebagai berikut :

- Perulangan merupakan struktur yang memungkinkan untuk menjalankan kode program yang sama beberapa kali dengan menggunakan suatu kondisi tertentu.
- Perulangan dalam javascript memiliki tiga bentuk, yakni perulangan for, while dan do-while.
- Inisialisasi disini digunakan sebagai bagian penentuan variabel yang akan digunakan untuk mengendalikan jumlah perulangan yang akan dilakukan.
- Hanya for yang memerlukan pernyataan inisialisasi dan peubah.

d. Tugas.

Buatlah program untuk menghitung nilai rata-rata, mencari nilai maksimal dan minimal dari suatu kumpulan nilai?

e. Tes Formatif.

1. Apa yang dimaksud dengan perulangan dalam program?
2. Sebutkan jenis-jenis perulangan yang kalian ketahui?
3. Bentuk perulangan mana yang paling baik dalam menampilkan data array menurut pendapat kalian, berikan alasannya?
4. Dari ketiga bentuk perulangan yang telah dijelaskan, manakah perulangan yang akan dijalankan jika kondisi awalnya tidak benar (false)? Berikan penjelasannya.
5. Mungkinkah menggunakan perulangan do-while tanpa tanda kurung kurawal?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

13. Kegiatan Belajar 13 : Memahami Penerapan Fungsi

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 13 ini siswa diharapkan dapat :

1) Mengetahui kode program fungsi
2) Memahami penerapan fungsi dalam pengembangan aplikasi

b. Uraian Materi.

Fungsi dalam pemrograman merupakan suatu mekanisme yang digunakan untuk mengelompokkan program dalam menyelesaikan suatu kasus, proses atau rumus aritmatika tertentu. Dengan pengertian yang sama pengelompokkan kode ini juga sering disebut subprogram (program bagian), prosedur ataupun metode. Dalam javascript semuanya itu disebut fungsi, karena Javascript hanya menyediakan satu kata kunci untuk membuat fungsi ini, yaitu function.

Fungsi dalam proses pemanggilannya (eksekusi) ada yang membutuhkan data tertentu agar dapat berjalan. Data tertentu yang perlu diberikan saat pemanggilan fungsi ini dikenal dengan nama parameter. Setiap fungsi dapat tidak memiliki atau memiliki lebih dari satu parameter.

Aturan penamaan fungsi mengikuti aturan penamaan variabel. Bentuk dari blok fungsi adalah sebagai berikut.

```
function nama_fungsi ( parameter1, parameter2, ...,
                       parameterN )
{
      kode program sebagai definisi fungsi
      return  nilai_balik_jika_ada;
}
```

Keterangan :

nama_fungsi: nama dari fungsi yang akan dibuat.

parameter1...N: merupakan syarat yang diperlukan oleh fungsi tersebut untuk dapat berjalan. Suatu fungsi juga dapat tidak memiliki parameter, artinya dapat dijalankan tanpa syarat.
return: perintah untuk membelikan hasil operasi dari fungsi ke user.
nilai_balik_jika_ada: nilai hasil operasi dari fungsi. Suatu fungsi dapat juga tidak memiliki nilai balik. Jika tidak memiliki nilai balik maka pernyataan return dapat ditiadakan.

Untuk menjalankan program yang ada dalam fungsi tersebut kita dapat memanggilnya dengan cara, sebagai berikut :

```
varHasil = nama_fungsi ( nilai_parameter1, nilai_parameter2,
..., nilai_parameterN ) ;
```

Keterangan :
varHasil: variabel yang dapat digunakan untuk menampung nilai balik dari fungsi yang dipanggil bila ada.

Dilihat dari hasil akhirnya fungsi dalam javascript dibedakan jadi dua, yakni fungsi yang memiliki nilai balik dan tidak. Nilai balik disini maksudnya adalah nilai hasil operasi yang dijalankan didalam fungsi diberikan ke bagian program lainnya untuk diproses lebih lanjut. Nilai balik dalam javascript dinyatakan dengan menggunakan kata kunci return. Contoh kasusnya dalam program penghitung luas dan volume limas. Dimana dalam perhitungan volume juga terdapat menyertakan perhitungan luas didalamnya. Dengan kata lain untuk menghitung volume limas kita harus menghitung luas alasnya terlebih dahulu. Dalam program, proses ini diterjemahkan sebagai berikut.

```
var panjang = 5;
var lebar = 3;
var tinggi = 4;
var luasAlas;
var volume;
```

```
luasAlas = panjang * lebar;
volume = luasAlas * tinggi / 2;
```

Apabila diubah dalam bentuk fungsi program diatas akan menjadi.

```
function hitungLuasAlas ( p, l ) {
      var luas = p * l;
      return luas;
}

function hitungVolume ( la, t ) {
      var vol = la * t / 2;
      return vol;
}

var panjang = 5;
var lebar = 3;
var tinggi = 4;
var luasAlas;
var volume;

luasAlas = hitungLuasAlas(panjang, lebar);
volume = hitungVolume(luasAlas, tinggi);

alert(luasAlas);
alert(volume);
```

Contoh : fungsi tanpa nilai balik.

```
function sapa(nama){
      alert("Halo, " + nama);
}
sapa(“bintang”);
```

Contoh fungsi tanpa parameter.

```
function sayHalo(){
      alert("Halo");
}
sayHalo();
```

PUSTAKA BAWAAN JAVASCRIPT

Fungsi alert(), prompt() dan parseInt() yang telah digunakan sebelumnya merupakan beberapa fungsi bawaan yang sudah disertakan dalam javascript. Fungsi tersebut menjadi bagian dari suatu kumpulan fungsi yang dikenal dengan Obyek. Ada banyak obyek dalam Javascript. Obyek ini tidak hanya memiliki fungsi namun juga memiliki konstanta-konstanta seperti PI pada obyek Math. Berikut ini beberapa obyek dasar yang dapat digunakan untuk mengolah data dalam javascript.

Obyek String

Obyek untuk mengolah data dalam bentuk alfanumerik (string) di Javascript.

Fungsi/Konstanta	Penjelasan
charAt()	Pengambilan satu karakter dari suatu string. Contoh: var nama = "Budi"; var posisiKarakter = 2; var k = nama.charAt(posisiKarakter); // k akan bernilai 'd'
indexOf()	Mencari posisi string dalam string tertentu. Contoh: var nama = "SMK Negeri"; var posNegeri = nama.indexOf("Negeri"); // posNegeri akan bernilai 4
lastIndexOf()	Mirip seperti indexOf(), fungsi ini mencari posisi terakhir dari kata yang ditentukan dalam suatu string tertentu, yang

	dapat saja mengandung lebih dari satu untuk kata yang dicari.
replace()	Mengganti kata tertentu dari suatu string. Contoh: var kalimat = "Java adalah bahasa skrip untuk web. Java bersifat multiplatform."; var kalimatBaru = kalimat.replace("Java", "Javascript");
split()	Memecah suatu string menjadi kumpulan string yang disimpan dalam format array. Contoh: var sms = "BUDI#17#Indonesia Raya"; var hasil = sms.split("#"); hasil akan berisi: hasil[0] berisi "BUDI" hasil[1] berisi "17" hasil[2] berisi "Indonesia Raya"
substr()	Mengambil kata tertentu dari suatu string. Contoh: var tglLahir = "17-08-1945"; var posisiAwal = 3; var jumlahKarakter = 2; var bulan = tglLahir.substr(posisiAwal, jumlahKarakter); // bulan akan berisi "08"
substring()	Mirip seperti substr() hanya saja parameter kedua diisikan dengan posisi karakter terakhir ditambah 1 dari kata yang mau diambil. Contoh: var tes = "Halo javascript"; var hasil = tes.substring(5, 9); // hasil akan berisi "java"
toLowerCase() toUpperCase()	Mengubah case dari suatu string. toUpperCase() akan membuat semua karakter berganti menjadi huruf kapital semua, demikian juga sebaliknya dengan toLowerCase().
trim()	Menghilangkan karakter kosong (spasi, tab, enter dan

	sejenisnya) di awal dan akhir suatu string. Contoh: var tes = " Java script "; var hasil = tes.trim(); // hasil akan bernilai "Java script"
length()	Menghitung jumlah karakter dalam suatu string. Contoh: var tes = " JS "; var hasil = tes.length(); // hasil akan bernilai 4

Obyek Math

Obyek untuk mengolah data dalam bentuk numerik di Javascript.

Fungsi/Konstanta	Penjelasan
E	Konstanta bilangan euler, sebesar 2.718
PI	Konstanta nilai PI, sebesar 3.14
sin(x) cos(x) tan(x) asin(x) acos(x) atan(x)	Mencari nilai sin, cos atau tan dari x dalam radian.
ceil(x)	Membulatkan nilai x keatas.
floor(x)	Membulatkan nilai x kebawah.
round(x)	Membulatkan nilai x.
log(x)	Mencari nilai log E dari x
pow(x, p)	Mencari nilai pangkat p dari x
random()	Menghasilkan bilangan random antara 0.0 - 1-0 Contoh: var acak = Math.random(); // acak dapat saja bernilai 0.0379
sqrt(x)	Mencari akar pangkat 2 dari x

Obyek Array

Obyek untuk mengolah data dalam bentuk kumpulan data (array) di Javascript.

Fungsi/Konstanta	Penjelasan
Length	Konstanta untuk mengetahui jumlah elemen dalam suatu array. Contoh: var d = [2.3, 4.5, 6.7, 8.9]; alert(d.length); // hasilnya: 4
join(penyambung)	Menggabungkan semua elemen dalam satu string khusus. Parameter penyambung menentukan string yang akan digunakan untuk menghubungkan setiap elemen yang ada dalam array, apabila tidak diisi maka tanda ‘,’ yang akan digunakan. Contoh: var d = [“bintang”, “aditya”, “cahaya”]; var s = d.join(); // hasilnya: “bintang,aditya,cahaya” var t = d.join(“ dan “); // “bintang dan aditya dan cahaya”
reverse()	Membalik urutan dari elemen-elemen array. Contoh: var d = [“bintang”, “aditya”, “cahaya”]; var s = d.reverse(); // hasilnya: [“cahaya”, “aditya”, “bintang”]
indexOf()	Mencari index dari suatu elemen dalam array. Contoh: var d = [“bintang”, “aditya”, “cahaya”]; var pos = d.indexOf(“aditya”); // hasilnya: 1
lastIndexOf()	Mencari index akhir dari suatu elemen dalam array. Prinsip kerjanya sama seperti indexOf() hanya saja, apabila ada lebih dari satu elemen yang ditemukan, yang diberikan adalah posisi elemen terakhirnya.

Referensi untuk kelas-kelas lainnya dapat dilihat pada web berikut.

http://www.w3schools.com/jsref/

c. Rangkuman.

Dari kegiatan belajar diatas dapat dibuat kesimpulan bahwa :

- Dalam Javascript kode program yang dihasilkan dapat dikelompok-kelompokkan menggunakan fungsi.
- Setiap fungsi dapat digunakan untuk menyelesaikan satu kasus, prose atau rumus aritmatika tertentu.
- Ada banyak obyek bawaan yang sudah dibuat oleh Javascript, diantaranya adalah String untuk pengolahan data alfanumerik, Math untuk menghitung rumus aritmatika, dan Array untuk mengolah data yang disimpan dalam array.

d. Tugas.

1. Buatlah fungsi untuk menghitung jumlah karakter, kata dan kalimat dalam satu data alfanumerik (string), dengan ketentuan sebagai berikut.

   Karakter disini adalah: huruf a-z (besar maupun kecil) dan angka 0-9
   Kata disini adalah: kumpulan karakter diatas tanpa spasi, ‘ ‘.
   Kalimat disini adalah: kumpulan kata yang diakhiri dengan tanda titik, ‘.’.

   Contoh penggunaannya:

   ```
   var data = “Kalimat 1. Kalimat 2. Kalimat 3”;
   var jmlKata = hitungJumlahKata(data);
   alert(jmlKata); // hasil: 6
   var jmlKalimat = hitungJumlahKalimat(data);
   alert(jmlKalimat); // hasil: 3
   var jmlKarakter = hitungJumlahKarakter(data);
   alert(jmlKarakter); // hasil: 24
   ```

2. Buatlah fungsi untuk membaca nilai bilangan bulat dari 0 hingga 20, seperti contoh kasus berikut:

   ```
   var d = bacaBilangan(2);
   alert(d); // hasilnya: dua
   var e = bacaBilangan(12);
   ```

```
alert(e); // hasilnya: dua belas
var f = bacaBilangan(20);
alert(f); // hasilnya: dua puluh
```

e. Tes Formatif.

1. Apa yang dimaksud dengan fungsi?
2. Mengapa pembuatan fungsi diperlukan dalam pengembangan program?
3. Sebutkan aturan-aturan terkait penamaan fungsi yang benar?
4. Apa yang dimaksud dengan parameter pada fungsi?
5. Mungkinkah kita memberikan pernyataan return pada fungsi yang tidak memiliki nilai balik? Jelaskan.

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

………

………

………

………

………

………

g. Lembar Kerja Siswa.

14. Kegiatan Belajar 14 : Pemrograman Berorientasi Obyek

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 14 ini siswa diharapkan dapat :

1) Mengetahui pemrograman berorientasi obyek pada web klien
2) Memahami penerapan pemrograman berorientasi obyek pada web klien

b. Uraian Materi.

Pemrograman Berorientasi Obyek (PBO)

Pemrograman berorientasi obyek adalah sebuah paradigma dalam pemrograman yang mengelompokkan fungsi, prosedur, variabel ataupun konstanta terkait dalam satu kesatuan (kelas) seperti obyek dalam dunia nyata. Obyek sendiri merupakan wujud dari kelas, secara umum dikenal dengan nama instan. Teknik-teknik dalam pemrograman ini merupakan pengembangan dari teknik sebelumnya seperti modularitas, polimorfisme, dan enkapsulasi. Saat ini ada banyak bahasa pemrograman yang telah mendukung pemrograman berorientasi obyek ini, seperti C++, C#, Java, Javascript, PHP, Python, Ruby, dan Objective C.

Dalam PBO setiap program akan menyatakan kumpulan dari obyek dan bukan lagi kumpulan fungsi ataupun kumpulan baris kode. Pada PBO ini setiap obyek memiliki kemampuan untuk menerima pesan, memproses data dan mengirim data ke obyek lainnya. Setiap obyek dapat dilihat sebagai program sendiri yang memiliki peran dan tanggungjawab masing-masing. Pengembangan program dengan menggunakan teknik PBO ini populer dalam program berskala besar karena fleksibel dan memudahkan dalam perbaikan.

PBO: Pemrograman Berbasis Prototipe

Javascript menggunakan PBO dengan model pemrograman berbasis prototipe. Model ini tidak mengenal kelas, yang ada adalah obyek. Obyek dalam model ini dapat dimodifikasi secara runtime seperti penambahan variabel (properti) ataupun metode (fungsi/prosedur). Model ini juga dikenal dengan istilah pemrograman PBO tanpa kelas, berorientasi prototipe atau berbasis instan.

Javascript Object Oriented Programming

Obyek Utama (Bawaan)

Javascript memiliki banyak obyek bawaan seperti Math, Array, String dan Object. Setiap obyek dalam Javascript adalah instan dari obyek Object, sehingga memiliki semua properti dan metode yang ada.

Obyek Buatan

Javascript merupakan bahasa PBO yang menggunakan kata kunci function untuk mendefinisikan suatu kelas. Oleh karena itu, pembuatan kelas sama dengan pembuatan fungsi dalam Javascript. Berikut ini merupakan format yang dapat digunakan untuk mendefinisikan kelas pada javascript.

```
function NamaKelas () {
}
```

Atau dalam bentuk yang lebih lengkap

```
function NamaKelas (argumen1, argumen2, …, argumenN)
{
      this.namavar1 = nilaiInisial;
      this.namavar2 = nilaiInisial;
      …

      this.namaFungsi1 = function(parameter1, parameter2, …,
parameterN){
            …
}
…
}
```

Hal yang dapat diperhatikan pada format diatas adalah namavar1 dan namavar2 merupakan properti dari kelas tersebut. Pembuatan properti dari suatu obyek dilakukan dengan menambahkan kata kunci this di awal dari nama properti

tersebut, seperti diperlihatkan dalam format diatas. Sedangkan pembuatan metode atau fungsi dari obyek dilakukan dengan cara yang sama hanya dengan properti dengan menambahkan definisi fungsinya dengan menggunakan kata kunci function.

Pembuatan obyek dari kelas yang telah didefinisikan diatas dapat dilakukan dengan menggunakan format berikut.

```
var  namaObyek =  new NamaKelas ( ) ;
```

Kemudian untuk mengakses properti ataupun metode yang dimiliki oleh suatu obyek dapat dengan menggunakan operator titik (.) setelah nama obyek diikuti nama properti atau metodenya. Mengenai aturan penamaan kelas dapat mengikuti aturan penamaan fungsi pada materi sebelumnya, contohnya ditunjukan dibawah ini.

```
<script>
function SegiEmpat(){
      this.panjang = 0;
      this.lebar = 0;

      this.hitungLuas = function(){
            var luas = this.panjang * this.lebar;
            return luas;
      }

      this.hitungKeliling = function(){
            var kel = 2 * (this.panjang + this.lebar);
            return kel;
      }
}

// contoh pengunaannya
var s = new SegiEmpat();
```

```
s.panjang = 4;
s.lebar = 3;
alert(s.hitungLuas());
alert(s.hitungKeliling());
</script>
```

Apabila pembuatan dilakukan dengan menggunakan format yang pertama, bentuk definisinya akan seperti berikut.

```
<script>
function SegiEmpat(){ }

SegiEmpat.prototype.panjang = 0;
SegiEmpat.prototype.lebar = 0;

SegiEmpat.prototype.hitungLuas = function(){
      var luas = this.panjang * this.lebar;
      return luas;
}

SegiEmpat.prototype.hitungKeliling = function(){
      var kel = 2 * (this.panjang + this.lebar);
      return kel;
}

// contoh pengunaannya
var s = new SegiEmpat();
s.panjang = 4;
s.lebar = 3;
alert(s.hitungLuas());
alert(s.hitungKeliling());
</script>
```

Dari contoh diatas dapat dilihat ada penggunaan kata kunci prototype pada obyek yang dibuat untuk menambahkan properti ataupun metodenya.

c. Rangkuman.

Pemrograman berorientasi obyek merupakan teknik pemrograman yang melihat suatu program sebagai kumpulan obyek yang memiliki properti dan metode yang saling berinteraksi satu sama lain melalui pengiriman pesan (data). Javascript merupakan bahasa yang tidak mengenal kelas atau bahasa PBO yang berorientasi prototipe atau instan. Dengan model ini memungkinkan program Javascript untuk mengolah obyek-obyek yang ada secara langsung, tanpa perlu mendefinisikan kelasnya. Javascipt menggunakan kata kunci function untuk mendefinisikan suatu obyek dan kata kunci property untuk memodifikasi suatu obyek dengan menambahkan atau mengubah definisi satu metode atau properti.

d. Tugas.

Buatlah program menggunakan teknik PBO untuk menyelesaikan perhitungan volume bangun ruang 3 dimensi berikut.





Gambar dari wikipedia.org (piramid) dan kompasiana.com (honai).

Ukuran dari masing-masing bangun ruang dapat ditentukan sendiri.

e. Tes Formatif.

1. Apa yang dimaksud dengan pemrograman berorientasi obyek?
2. Jelaskan perbedaan PBO pada Javascript dengan bahasa PBO lainnya seperti C++ dan Java?
3. Apa perbedaan antara PBO dengan pemrograman prosedural?

4. Apa yang dimaksud dengan prototipe pada PBO?
5. Apa manfaat yang diperoleh dari penerapan PBO dalam pengembangan program berskala besar?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

15. Kegiatan Belajar 15 : Mengolah Interaksi User

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 15 ini siswa diharapkan dapat :

1) Mengetahui bentuk penanganan interaksi user pada aplikasi
2) Memahami penerapan teknik pengolahan interaksi user pada aplikasi

b. Uraian Materi.

Interaksi User

Adanya interaktifitas dalam aplikasi akan membantu menarik minat pengguna untuk menggunakan aplikasi yang dikembangkan. Di banyak bahasa pemrograman pengolahan interaksi ini dikenal sebagai teknik penanganan event. Event dalam Javascript adalah sebuah kejadian interaksi antara user dengan halaman web. Bentuk interaksi ini dapat digambarkan seperti pada saat user melakukan klik pada tombol ataupun link, mengetik username dan password dan sebagainya.

Event ini pada HTML ditambahkan dalam bentuk atribut dari elemen yang ada. Program javascript yang ditempatkan pada atribut tersebut akan dijalankan pada saat interaksi yang bersesuaian terjadi. Berikut ini merupakan daftar atribut yang dapat digunakan untuk mengolah interaksi user.

Tabel 1. Daftar atribut event yang sering ditemui dalam dokumen HTML

Atribut Event	Penjelasan
onclick ondblclick	Event ini terjadi saat user melakukan klik sekali atau dua kali pada satu elemen HTML Elemen HTML yang umum memiliki atribut ini: button, document, checkbox, link, radio, select, submit, reset.
onmousedown	Event ini terjadi saat user melakukan klik pada satu elemen HTML Elemen terkait: button, document, link, layer
onmousemove	Event ini terjadi saat user menggerakan mouse diatas elemen HTML

	Elemen terkait: button, document, link, layer
onmouseover	Event ini terjadi saat user menggerakan mouse ke atas elemen HTML Elemen terkait: button, document, link, layer
onmouseout	Event ini terjadi saat user menggerakan mouse keluar elemen HTML Elemen terkait: button, document, link, layer
onmouseup	Event ini terjadi saat user melepaskan klik pada elemen HTML Elemen terkait: button, document, link, layer
onkeydown	Event ini terjadi saat user sedang menekan tombol keyboard pada elemen HTML Elemen terkait: document, image, link, input, textarea
onkeypress	Event ini terjadi saat user menekan tombol keyboard pada elemen HTML Elemen terkait: document, image, link, input, textarea
onkeyup	Event ini terjadi saat user melepas penekanan tombol keyboard pada elemen HTML Elemen terkait: document, image, link, input, textarea
onload	Event ini terjadi dokumen atau frame telah ditampilkan oleh browser. Elemen terkait: document, image, layer, window
onresize	Event ini terjadi saat ukuran dari dokumen berubah Elemen terkait: document, image, layer, window
onscroll	Event ini terjadi saat user melakukan scrolling pada dokumen Elemen terkait: document, window
onunload	Event ini terjadi saat user menutup dokumen Elemen terkait: document, window
onblur	Event ini terjadi saat fokus pada elemen tersebut dipindahkan ke elemen lainnya Elemen terkait: button, checkbox, file upload, layer, password, radio, reset, select, submit, text, textarea, window
onchange	Event ini terjadi saat nilai elemen form berubah Elemen terkait: file upload, select, submit, text, textarea

onfocus	Event ini terjadi saat elemen tersebut difokuskan oleh user Elemen terkait: button, checkbox, file upload, layer, password, radio, reset, select, submit, text, textarea, window
onreset	Event ini terjadi saat form di reset ke kondisi sebelum diisi Elemen terkait: form
onselect	Event ini terjadi saat user memiliki sebagian teks pada elemen Elemen terkait: text, textarea
onsubmit	Event ini terjadi saat form akan dikirim ke aplikasi server Elemen terkait: form

Format penerapan kode Javascript pada atribut event diatas sebagai berikut :

```
<tag atribut-event=”kode javascript;” … > … </tag>
```

atau untuk tag tanpa penutup.

```
<tag atribut-event=”kode javascript;” … />
```

Contoh program penerapan event pada halaman web.

```
<a href=”http://kompas.com” onclick=”alert(‘Link ini akan
membuka halaman Kompas.com’);”>Link 1</a>
```

Pada contoh diatas, apabila user melakukan klik pada Link 1, maka akan ditampilkan alert(). Selanjutnya pada contoh berikut ini sudah melibatkan fungsi di Javascript. Dimana apabila nilai dari input text berubah, maka alert() akan tampil dengan mengambil nilai yang diketikkan oleh user.

```
<script>
function sayHalo(){
     var nm = document.getElementById('nama').value;
     alert("Halo, " + nm);
}
</script>
<input type="text" id="nama" onchange="sayHalo();" />
```

Pernyataan document.getElementById('id elemen').value adalah perintah javascript untuk mengambil nilai input dari elemen form.

Pengaksesan Obyek Pada Halaman Web

Setiap elemen HTML yang ada dalam suatu dokumen web oleh Javascript diterjemahkan sebagai obyek HTML. Pada obyek HTML ini terdapat sejumlah fungsi, konstanta ataupun variabel yang dapat diakses untuk mengubah elemen HTML tersebut. Akses terhadap obyek HTML tersebut dapat dilakukan dengan menggunakan fungsi-fungsi berikut.

Tabel 2. Fungsi untuk mengakses obyek HTML

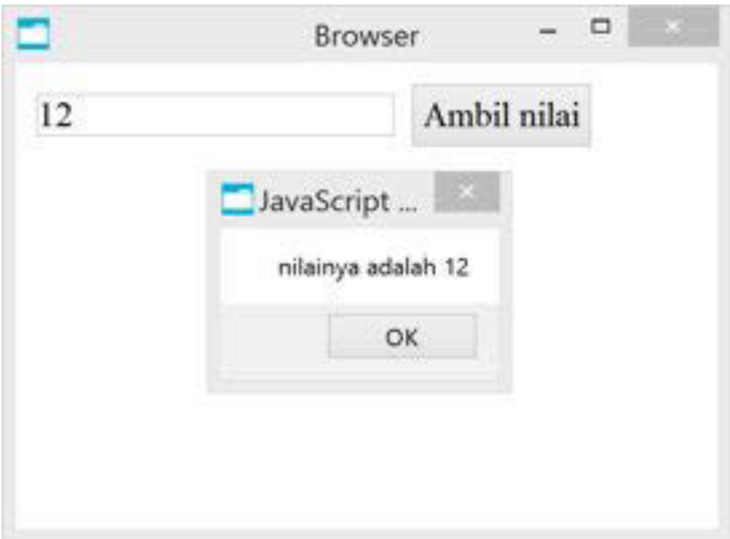
Fungsi	Penjelasan
getElementById("id-elemen")	id-elemen disini digunakan untuk menunjuk nilai dari atribut id dari elemen yang mau diakses obyeknya.
getElementsByTagName("nama-tag")	nama-tag disini digunakan untuk mengambil obyek berdasarkan nama tag HTML-nya
getElementsByName("nama")	nama disini adalah nilai dari atribut name dari elemen yang akan diakses. Umumnya diterapkan pada komponen dari form, seperti input, select, checkbox, dan komponen lainnya.

Ketiga fungsi diatas merupakan bagian dari obyek document. Sehingga obyek document perlu disertakan pada saat memanggil fungsi diatas.

Contoh : penggunaan fungsi getElementById()

```
<input type="text" id="nilai" />
<input type="button" id="btnAmbilNilai" value="Ambil nilai"
onclick="ambilNilai();" />
<script>
function ambilNilai(){
     var objNilai = document.getElementById('nilai');
     var n = objNilai.value;
     alert("nilainya adalah " + n);
}
</script>
```

Pada contoh diatas ini apabila tombol Ambil nilai diklik, maka obyek input nilai akan diambil menggunakan fungsi document.getElementById() dan selanjutnya, variabel value dari obyek nilai digunakan untuk mengambil data masukkan dari user agar dapat ditampilkan.



Gambar 14.1 Contoh hasil penggunaan fungsi getElementById()

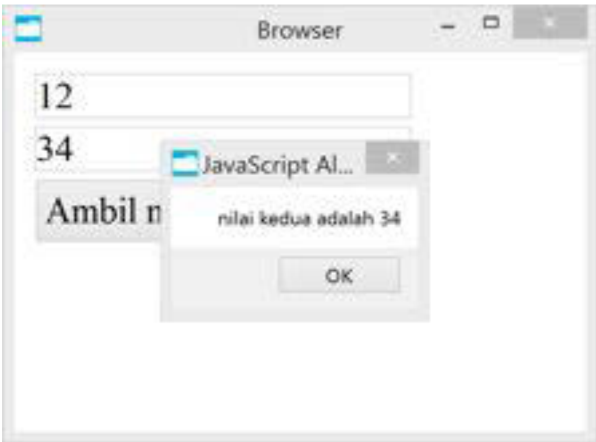
Contoh : penggunaan fungsi getElementsByName().

```
<input type="text" name="nilai" />
<input type="button" id="btnAmbilNilai" value="Ambil nilai"
onclick="ambilNilai();" />
<script>
function ambilNilai(){
     var objNilai = document.getElementsByName('nilai')[0];
     var n = objNilai.value;
     alert("nilainya adalah " + n);
}
</script>
```

Fungsi getElementsByName() sedikit berbeda dengan metode getElementById(), dimana hasil dari metode ini adalah array obyek bukan obyek tunggal. Dengan hasil berupa array ini berarti untuk mengakses elemen pertama digunakan index 0. Disisi lain, dengan hasil array ini dimungkinkan untuk menuliskan elemen HTML dengan nilai atribut name yang sama. Berikut ini merupakan contoh penggunaan fungsi getElementsByName() dengan elemen yang memiliki nama sama lebih dari satu.

```
<input type="text" name="nilai" />
<input type="text" name="nilai" />
<input type="button" id="btnAmbilNilai" value="Ambil nilai"
onclick="ambilNilai();" />
<script>
function ambilNilai(){
     var obj1 = document.getElementsByName('nilai')[0];
     var obj2 = document.getElementsByName('nilai')[1];
     var n1 = obj1.value;
     var n2 = obj2.value;
     alert("nilai pertama adalah " + n1);
     alert("nilai kedua adalah " + n2);
}
</script>
```

Hasilnya:



Gambar 14.2 Contoh hasil penggunaan fungsi getElementsByName()

Selanjutnya untuk fungsi getElementsByTagName(), berikut ini diberikan contoh skripnya.

```
<input type="text" name="nilai" />
<input type="button" id="btnAmbilNilai" value="Ambil nilai"
onclick="ambilNilai();" />
<input type="text" name="hasil" />
<script>
function ambilNilai(){
     // mengambil obyek input text pertama
     var obj1 = document.getElementsByTagName('input')[0];

     // mengambil obyek input button
     var obj2 = document.getElementsByTagName('input')[1];

     // mengambil obyek input text kedua
     var obj3 = document.getElementsByTagName('input')[2];

     // mengambil nilai input text pertama dan button
     var s1 = obj1.value;
     var s2 = obj2.value;
```

```
    // menampilkan nilai pada input text kedua
    obj3.value = s1 + s2;
}
</script>
```

Hasil akhir dari skrip ini mirip dengan contoh sebelumnya, hanya cara mengambil nilainya yang berbeda. Oleh karena tombol juga dibuat menggunakan tag `<input>` maka nilainya dapat diakses menggunakan fungsi ini. Berikutnya untuk obyek input text yang kedua digunakan untuk menampilkan isi dari input text pertama dan input button.



Gambar 14.3 Contoh hasil penggunaan fungsi getElementsByTagName()

c. Rangkuman.

Dari kegiatan belajar diatas dapat dibuat kesimpulan bahwa :

- Event merupakan konsep penting dalam pemrograman, karena dari konsep ini aplikasi dapat berinteraksi dengan user melalui berbagai input yang ada.
- Event pada javascript dapat diterapkan pada semua elemen yang ada. Jenis interaksi yang diberikan oleh user adalah melalui perangkat input berupa mouse dan juga keyboard.

- Dalam Javascript sudah disediakan sejumlah fungsi untuk mengakses data yang diberikan oleh user melalui elemen HTML, diantaranya adalah getElementById(), getElementsByName() dan getElementsByTagName(), yang kesemuanya merupakan bagian dari obyek document. Selanjutnya dapat diolah sesuai hasil yang diharapkan.

d. Tugas.

1. Lakukan percobaan menggunakan fungsi pengaksesan diatas untuk mengambil nilai dari beberapa obyek HTML dan simpan hasilnya dalam tabel.
   a. Radio untuk memilih jenis kelamin L atau P.
   b. Checkbox untuk memilih warna kesukaan merah, jingga, kuning, hijau, biru, dan ungu.
   c. Select untuk memilih golongan darah
   d. Textarea untuk entri data alamat

   Lanjutkan jawaban dibawah ini untuk komponen lainnya.

   Radio

```
<input type="radio" id="kelaminL" name="kelamin"
value="L" /> Laki-laki<br />
<input type="radio" id="kelaminP" name="kelamin"
value="P" /> Perempuan<br />
<input type="submit" value="Get"
onclick="ambilNilai();" />

<script>
function ambilNilai(){
     var objKelamin =
document.getElementsByName('kelamin');
     if(objKelamin[0].checked == true){
          alert("Kelaminnya adalah " +
objKelamin[0].value);
     }
```

```
     else {
          alert("Kelaminnya adalah " +
objKelamin[1].value);
     }
}
</script>
```

Select

```
...
```

Obyek HTML	Fungsi/Variabel yang diakses	Penjelasan dan hasil
Radio	checked	Untuk mengetahui apakah radio tersebut dipilih atau tidak. Hasil: True, jika dipilih False, jika tidak dipilih
	value	Untuk mengambil nilai dari radio. Hasil: L/P, sesuai pilihan user.
…	…	…

2. Buatlah suatu program untuk menghitung luas dari panjang dan lebar yang diberikan oleh user pada input text?
3. Buatlah program perhitungan index berat badan menggunakan javascript?
4. Buatlah program untuk menampilkan jumlah karakter dan kata yang diketikkan oleh user pada sebuah input textarea?

e. Tes Formatif.

1. Apakah yang dimaksud dengan event?

2. Apakah kaitan antara interaktifitas suatu aplikasi dengan event di Javascript?
3. Jelaskan metode-metode yang dapat digunakan untuk mengakses obyek HTML di Javascript?
4. Jelaskan event-event yang berkaitan dengan keyboard?
5. Apakah kelebihan dan kekurangan dari penggunaan fungsi getElementById() yang dapat memanggil obyek tunggal dan bukan array?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

16. Kegiatan Belajar 16 : Navigasi Halaman Web

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 16 ini siswa diharapkan dapat :

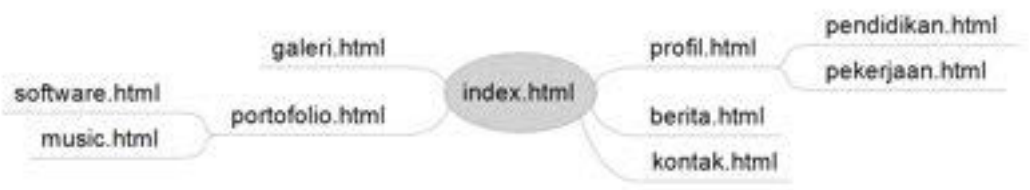
1) Mengetahui sistem navigasi halaman web melalui kode program
2) Memahami penerapan program navigasi pada halaman web

b. Uraian Materi.

Navigasi Halaman

Navigasi secara umum dapat digunakan untuk menunjukkan perpindahan dari satu titik ke titik yang lain. Dalam hal ini perpindahan yang dimaksud adalah perpindahan dalam satu halaman ataupun antar halaman web. Wujud dalam halaman web yang tampak adalah dalam bentuk link (tekstual), tombol dan gambar (grafis). Selain menggunakan HTML, navigasi pada halaman web juga dapat diwujudkan dengan menggunakan program Javascript.

Perancangan navigasi untuk suatu website dapat juga dikenal dengan istilah pembuatan sitemap. Penggambarannya dilakukan dengan terlebih dahulu menentukan halaman utama web, kemudian menentukan link-link apa saja dari yang ada di halaman tersebut dan mengaitkannya. Berikut ini merupakan salah satu contoh bentuk rancangan sitemap.



Gambar 15.1 Contoh sitemap sebuah website

Pembuatan sitemap ini dapat dilakukan dengan menggunakan aplikasi peta pikiran seperti Freemind, XMind ataupun sejenisnya. Sitemap ini nantinya dapat dijadikan panduan ataupun arah bagi programmer web dalam mengembangkan aplikasinya. Sedangkan untuk user dapat memberikan gambaran secara menyeluruh tentang web yang sedang dikunjungi.

Terdapat sejumlah metode yang dapat digunakan di Javascript untuk berpindah halaman ini, diantaranya adalah :

Teknik navigasi	Penjelasan
window.location.href = “url”;	Digunakan untuk menentukan halaman yang akan ditampilkan pada browser, melalui string url yang diberikan.
window.location.assign(“url”);	URL halaman baru diberikan dalam bentuk parameter untuk fungsi ini.
window.location.replace(“url”);	Mirip seperti assign(), bedanya browser tidak akan mengingat url sebelum ini, sehingga tidak akan ada perubahan apabila menekan back pada browser atau menjalankan metode history.back() untuk kembali ke halaman sebelumnya. Jadi apabila menginginkan halaman sebelumnya tetap ada gunakan fungsi assign().
window.history.back()	Fungsi untuk menampilkan kembali halaman sebelumnya. Sama fungsinya dengan tombol back pada browser.
window.history.forward()	Untuk menampilkan halaman setelah saat ini apabila ada.

Contoh:

File: index.html

```
 <h1>Halaman Index</h1>

 <ul>
 <li
onclick="window.location.href='profil.html'">Profil</li>
```

```
 <li
onclick="window.location.assign('portofolio.html');">Portofo
lio</li>
 <li
onclick="window.location.replace('kontak.html');">Kontak</li
>
 </ul>
```

File: profil.html

```
 <h1>Halaman Profil</h1>

 <ul>
 <li onclick="window.history.back();">Kembali</li>
 </ul>
```

File: portofolio.html

```
 <h1>Halaman Portofolio</h1>

 <ul>
 <li onclick="window.history.back();">Kembali</li>
 </ul>
```

File: kontak.html

```
 <h1>Halaman Kontak</h1>

 <ul>
 <li onclick="window.history.back();">Kembali</li>
 </ul>
```

Program diatas ini merupakan salah satu bentuk hasil dari sitemap yang dirancang diatas. Dimulai dari index.html berisikan link menu untuk mengakses halaman lainnya. Terdapat tiga fungsi navigasi yang diterapkan dihalaman ini. Kemudian masing-masing halaman profile.html, portofolio.html dan kontak.html terdapat menu untuk kembali ke halaman index.html. Hal yang membedakan

ketiga halaman tersebut adalah hasil dari pemilihan menu Kembali. Pada file profile.html dan portofolio.html ketika menu Kembali dipilih browser akan membuka halaman index.html. Hal ini karena browser merekam sejarah navigasi halaman sebelumnya. Sedangkan pada file kontak.html menu Kembali yang dipilih tidak akan membawa user ke halaman index.html. Ini dikarenakan rekaman jejak sejarah file index.html dihapus pada saat pemanggilan fungsi window.location.replace() oleh menu Kontak di file index.html.

NAVIGASI DENGAN AJAX (MATERI TAMBAHAN)

Bentuk lain dari navigasi ini adalah dengan menggunakan obyek AJAX pada Javascript. AJAX, Asynchronous Javascript and XML. AJAX bukan merupakan bahasa pemrograman baru, melainkan kumpulan kelas Javascript yang dapat digunakan salah satunya untuk menampilkan halaman lain tanpa merubah keseluruhan halaman saat ini.

Dengan obyek ini perpindahan halaman dapat dilakukan tanpa menghilangkan halaman yang sedang dibuka saat ini. Ini perbedaan antara menampilkan halaman dengan AJAX dan cara sebelumnya. Dimana pada saat menunggu halaman yang akan ditampilkan, halaman saat ini tetap terlihat, karena prosesnya terjadi dilatar dan tidak terlihat oleh user.

jQuery merupakan salah satu librari yang dapat digunakan untuk menerapkan AJAX ini dengan perintah-perintah yang lebih sederhana. Librari disini maksudnya adalah kumpulan kelas atau obyek yang didalamnya juga berisi fungsi, variabel dan konstanta. Dengan menyertakan librari ini dalam program akan dapat menambah kemampuan aplikasi sebagaimana fitur yang ditawarkan dalam librari tersebut.

Website http://jquery.com berisi link untuk mendownload librarinya dan juga ada banyak tutorial yang diberikan yang dapat diikuti untuk penerapan jQuery pada pemrograman halaman web. Sampai saat buku ini ditulis jQuery saat ini telah sampai pada versi 2.0.3.

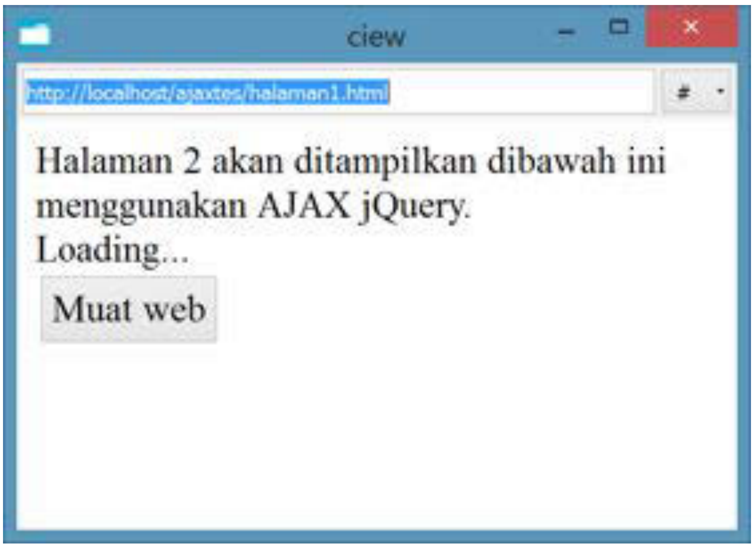
Berikut ini merupakan contoh penerapan librari jQuery untuk menampilkan halaman lain menggunakan AJAX. Contoh program :

File: halaman1.html

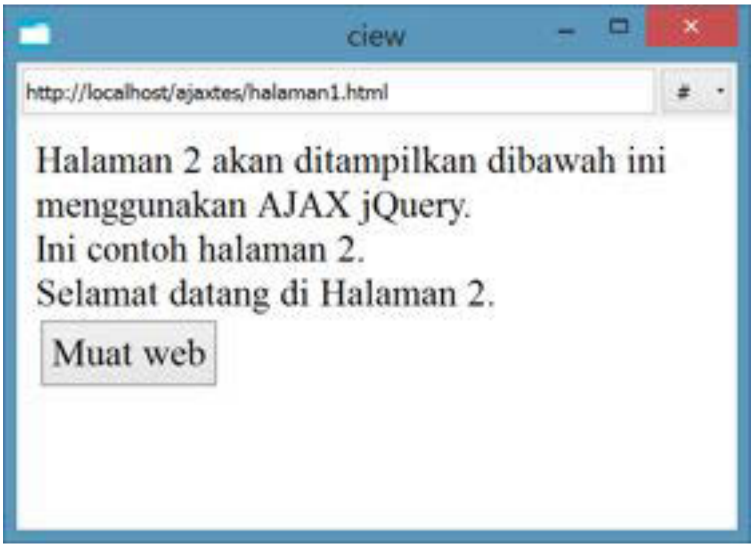
```
<div>Halaman 2 akan ditampilkan dibawah ini menggunakan
AJAX
jQuery.</div>
<div id="hal2">Loading...</div>
<button onclick="tampilkanWeb()">Muat web</button>
<script src="jquery-2.0.3.js"></script>
<script>
function tampilkanWeb(){
    $('#hal2).load("halaman2.html");
}
</script>
```

File: halaman2.html

```
Ini contoh halaman 2.<br>
Selamat datang di Halaman 2.<br>
```

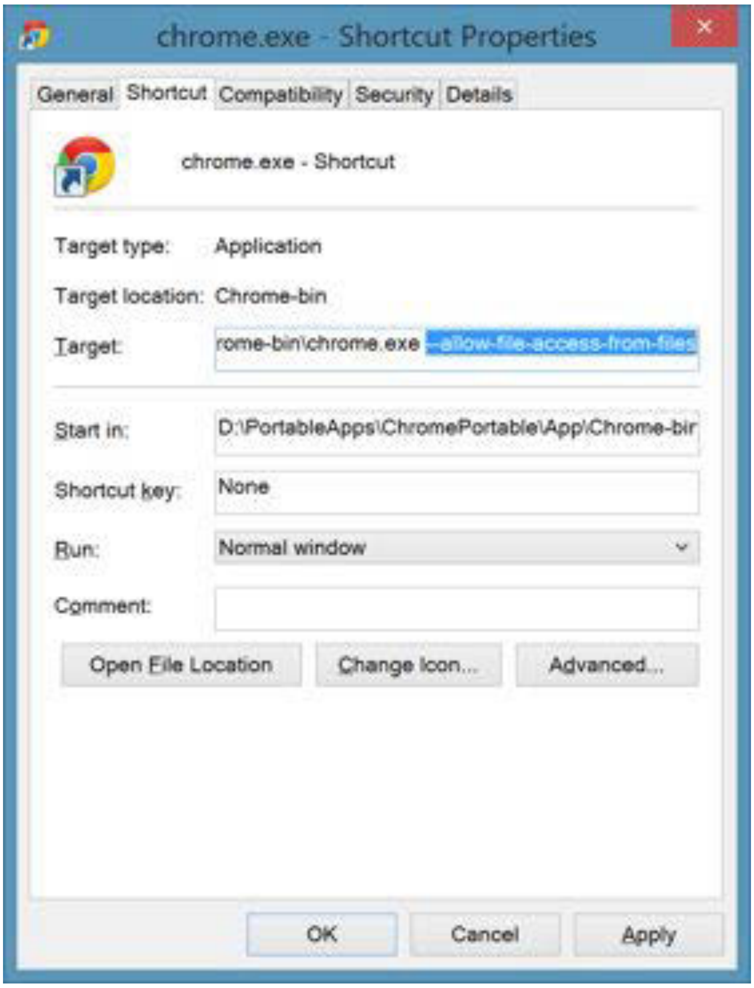


Gambar 15.2 Tampilan halaman1.html yang memuat program AJAX jQuery



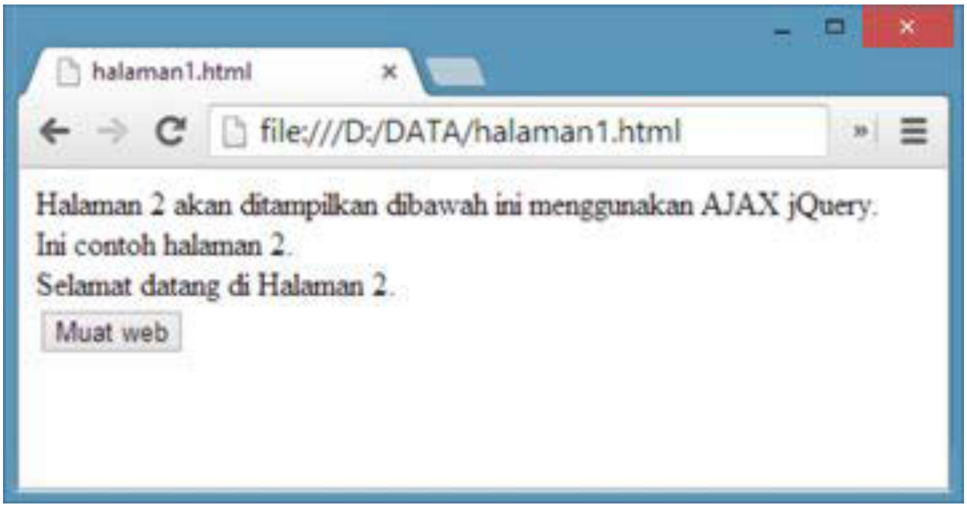
Gambar 15.3 Tampilan halaman 1. html setelah memuat halaman 2. html pada div dengan id hal2

Dari hasil percobaan ini dapat dilihat bahwa perubahan hanya terjadi pada tag `<div>` dengan id hal2 dan tidak pada alamat URL di browser, yang tetap berada pada halaman 1. html. Skrip ini dapat dijalankan apabila ditempatkan sebagai aplikasi server menggunakan web server seperti Apache (dengan menginstall aplikasi XAMPP atau sejenisnya) atau dengan menambahkan argumen --allow-file-access-from-files pada eksekusi browser Google Chrome seperti terlihat pada tampilan berikut.



Gambar 15.4 Penambahan argumen eksekusi pada browser Google Chrome

Setelah menambahkan argumen diatas, halaman HTML dapat dibuka kembali menggunakan browser Google Chrome yang sudah dimodifikasi ini. Hasilnya jika berhasil menambahkan argumen eksekusi diatas adalah sebagai berikut.



Gambar 15.5 Halaman dengan librari AJAX pada browser Google Chrome

c. Rangkuman.

Dari paparan kegiatan menerapkan navigasi halaman web dapat dibuat ringkasan materi sebagai berikut :

- Navigasi secara umum dapat digunakan untuk menunjukkan perpindahan dari satu titik ke titik yang lain.
- Selain menggunakan HTML, navigasi pada halaman web juga dapat diwujudkan dengan menggunakan program Javascript.
- Perancangan navigasi untuk suatu website dapat juga dikenal dengan istilah pembuatan sitemap.
- Sitemap ini nantinya dapat dijadikan panduan ataupun arah bagi programmer web dalam mengembangkan aplikasinya.
- Bagi user , sitemap dapat memberikan gambaran secara menyeluruh tentang web yang sedang dikunjungi.
- jQuery merupakan salah satu librari yang dapat digunakan untuk menerapkan AJAX ini dengan perintah-perintah yang lebih sederhana.

d. Tugas.

Buatlah website mengenai profil masing-masing siswa dengan minimal memiliki halaman utama, biodata, foto, portofolio, dan sitemap. Terapkan aturan berikut pada aplikasi web yang dihasilkan:

- Ada menu navigasi pada setiap halaman, termasuk juga untuk back dan forward.
- Menu navigasi ini dapat berupa link atau tombol.
- Kesemua item menu navigasi diterapkan menggunakan metode-metode navigasi yang telah dijelaskan termasuk AJAX.

e. Tes Formatif.

1. Apa yang dimaksud dengan sitemap?
2. Apa manfaat yang diperoleh user dengan adanya sitemap?
3. Apa perbedaan antara penerapan metode window.location.assign() dengan window.location.replace()?
4. Apa kelebihan dan kekurangan penerapan navigasi melalui skrip pada halaman web?
5. Apa yang dimaksud denga AJAX?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

………

………

………

………

LJ- 03 :

………

………

………

………

………

………

………

………

………

………

LJ- 04 :

………

………

………

………

………

………

………

………

………

LJ- 05 :

………

………

………

....
....
....
....
....
....

g. Lembar Kerja Siswa.

17. Kegiatan Belajar 17 : Format Dinamis Halaman Web

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 17 ini siswa diharapkan dapat :

1) Mengetahui teknik membuat format halaman web lebih dinamis
2) Memahami penerapan teknik pembuatan halaman web klien dinamis

b. Uraian Materi.

Format Dinamis Dengan Javascript

Selain karena adanya interaksi aplikasi web juga dapat lebih menarik apabila dilengkap dengan kemampuan untuk visualisasi setiap interaksi yang terjadi. Penerapan visualisasi interaksi ini dapat dilakukan dengan menggunakan CSS. Selain dapat digunakan untuk memodifikasi elemen HTML, Javascript juga dapat digunakan untuk mengubah style dari suatu elemen HTML. Pengubahan style ini dilakukan dengan menerapkan metode :

```
obyectHTML.style.properti = “style baru”;
```

obyekHTML disini merupakan obyek yang dihasilkan dari pemanggilan fungsi getElementById(), getElementsByName() atau getElementsByTagName(). Sedangkan properti disini menunjukkan nama style CSS yang akan dimodifikasi dari obyek tersebut. Berikut ini beberapa daftar properti yang dapat digunakan.

Properti	Penjelasan
color	Menentukan warna dari elemen. Alternatif nilainya: a. format hexa (#XXXXXX), contoh: “#121212” b. nama warna, contoh: “blue” c. format desimal (rgb(r, g, b)), contoh: “rgb(100, 20, 200)”
backgroundColor	Menentukan warna latar belakang dari elemen. Alternatif nilainya sama seperti color diatas.
visibility	Menentukan status visual dari suatu elemen. Nilainya: “visible” / “hidden”

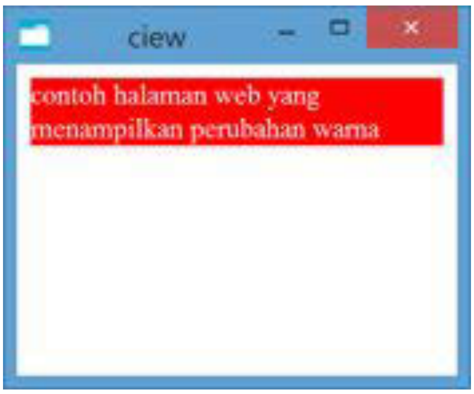
width	Menentukan lebar dari suatu elemen. Alternatif nilainya: d. auto e. inherit f. persentase, contoh: “56%” g. pixel, contoh: “100px”
height	Menentukan tinggi dari suatu elemen. Pemberian nilainya sama seperti properti width.

Daftar yang lebih lengkap untuk properti style yang dapat dimodifikasi dapat mengunjungi link berikut http://www.w3schools.com/jsref/dom_obj_style.asp. Contoh programnya diberikan sebagai berikut :

Pengubahan warna

```
<p style="background-color: orange; color: white">contoh halaman web yang menampilkan perubahan warna</p>

<script>
var par = document.getElementsByTagName("p")[0];
par.style.backgroundColor = "red";
</script>
```



Gambar 1. Perubahan warna latar suatu paragraf melalui Javascript

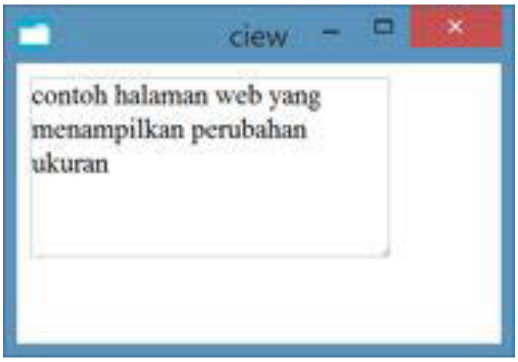
Pada contoh diatas dapat dilihat bahwa paragraf yang sebelumnya memiliki warna latar oranye diganti menjadi merah melalui skrip javascript.

Pengubahan ukuran teks area

```
 <textarea id="info" style="width: 100px; height: 100px;">contoh halaman web yang menampilkan perubahan ukuran</textarea>

 <script>
 var info = document.getElementById("info");
 info.style.width = "200px";
 </script>
```

Halaman web diatas ini apabila ditampilkan akan mengubah ukuran lebar dari obyek HTML dengan id info menjadi 200px. Perlu diketahui disini bahwa nilai 200px ditambahkan ke properti width dari style obyek info dengan menggunakan tanda kutip. Hasilnya seperti berikut ini.



Gambar 2. Hasil perubahan ukuran lebar dari obyek HTML melalui Javascript

Pengubahan penampakan suatu komponen

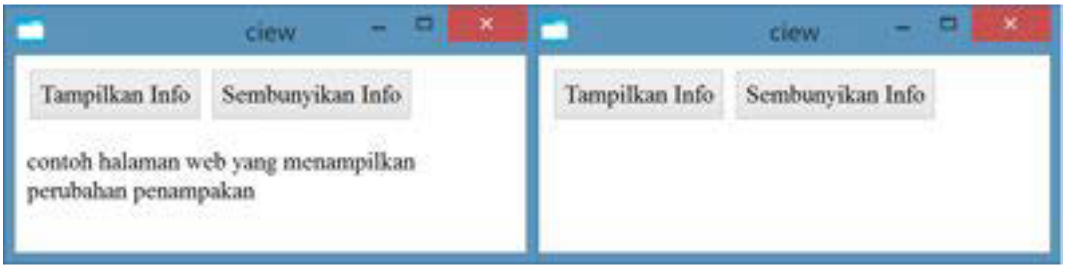
```
 <button onclick="tampilkanInfo()">Tampilkan Info</button>
 <button onclick="sembunyikanInfo()">Sembunyikan Info</button>

 <p id="info">contoh halaman web yang menampilkan perubahan penampakan</p>
```

```
<script>
function tampilkanInfo(){
     var info = document.getElementById("info");
     info.style.visibility = "visible";
}

function sembunyikanInfo(){
     var info = document.getElementById("info");
     info.style.visibility = "hidden";
}
</script>
```

Gambar dibawah ini merupakan gambar yang menunjukkan bahwa apabila tombol Sembunyikan Info ditekan akan menyebabkan informasi pada paragraf akan disembunyikan seperti ditunjukan pada gambar bagian kanan, demikian juga sebaliknya apabila tombol Tampilkan Info ditekan.



Gambar 3. Perubahan penampakan obyek HTML melalui Javascript

c. Rangkuman.

Javascript selain dapat digunakan untuk memodifikasi data-data yang tersimpan pada obyek HTML dapat juga digunakan untuk memodifikasi tampilan dari obyek tersebut. Ini dapat dilakukan melalui properti style yang dimiliki oleh setiap obyek HTML yang diakses melalui Javascript.

d. Tugas.

1. Kembangkan program pengubah penampakan pada contoh diatas sehingga apabila tombol **Sembunyikan Info** ditekan, tombol **Tampilkan**

Info ditampilkan dan tombol **Sembunyikan Info** disembunyikan juga selain menyembunyikan paragraf. Demikian juga sebaliknya untuk tombol **Tampilkan Info**.

2. Buatlah program untuk memilih warna secara acak dari suatu kumpulan warna dan menerapkan warna tersebut pada teks yang juga dipilih secara acak dari suatu kumpulan teks. Contoh hasilnya dapat dilihat pada gambar berikut.



e. Tes Formatif.

1. Apa yang dimaksud dengan format dinamis pada halaman web klien?
2. Sebutkan properti-properti yang dapat diakses pada style dari suatu obyek HTML?

3. Apa perbedaan antara properti style pada Javascript dengan CSS yang dapat diamati?
4. Mungkinkah memodifikasi style dari suatu obyek HTML menggunakan CSS yang dilakukan melaliu Javascript? Menggunakan kode CSS yang diterapkan melalui Javascript ke suatu obyek HTML.

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

LJ- 03 :

....

....

....

....

....

....

....

....

....

....

LJ- 04 :

....

....

....

....

....

....

....

....

....

LJ- 05 :

....

....

....

....

....

....

....

....

....

g. **Lembar Kerja Siswa.**

18. Kegiatan Belajar 18 : Penerapan Pustaka Eksternal

a. Tujuan Pembelajaran.

Setelah mengikuti kegiatan belajar 18 ini siswa diharapkan dapat :

1) Mengetahui pustaka eksternal pada skripting halaman web
2) Memahami penerapan pustaka eksternal pada halaman web

b. Uraian Materi.

Penerapan Pustaka Eksternal

Selain menyediakan pustaka bawaan, pada pemrograman Javascript juga dimungkinkan untuk menggunakan pustaka dari pengembang (*programmer*) Javascript lainnya. Pustaka eksternal sebenarnya adalah program Javascript seperti umumnya yang dikembangkan oleh perorangan atau instansi agar dapat digunakan oleh orang atau instansi lainnya. Ada banyak pustaka Javascript yang telah dikembangkan, seperti Dojo, Ext JS, YUI, dan jQuery. Bentuknya ada yang bersifat terbuka dan tertutup tergantung dari pengembangnya. Adanya penambahan pustaka eksternal ini akan dapat menambah fitur dari halaman web yang dihasilkan. Selain itu juga dapat memberikan kemudahan dalam pengembangan karena adanya penyederhanaan sejumlah operasi umum dalam Javascript, seperti fungsi-fungsi untuk mengakses obyek ataupun menjaga kompatibilitas antar browser.

Berikut ini format perintah yang dapat digunakan untuk menyertakan pustaka eksternal tersebut ke dalam halaman web.

<script src=”lokasi/file/pustaka.js”></script>

Sebagai contoh, apabila menggunakan jQuery dengan file pustakanya jquery/jquery.min.js dan berada pada lokasi yang sama dengan aplikasi webnya, maka penyertaannya dapat menggunakan perintah.

<script src=”jquery/jquery.min.js”></script>

Pada pembelajaran kali ini pembahasan pustaka eksternal akan difokuskan pada penerapan pustaka jQuery.

jQuery saat ini telah sampai pada rilis 1.11.0 untuk yang versi 1 dan 2.1.0 untuk versi 2. Pustaka ini dapat didownload pada alamat http://jquery.com. Ini merupakan salah satu pustaka yang banyak diterapkan saat ini dan memiliki sejumlah turunan, diantaranya adalah jQuery UI, jQuery Mobile dan Twitter Bootstrap. Perkembangan yang pesat dari jQuery ini dimungkinkan karena sifatnya yang terbuka. Keterbukaannya ini memungkinkan banyak programmer Javascript untuk berkontribusi mengembangkan fitur-fitur yang ada pada pustaka ini.

Melalui jQuery permasalahan kompatibilitas aplikasi antar web browser yang berbeda-beda dapat diatasi, seperti pada contoh kasus penerapan AJAX pada kegiatan belajar sebelumnya. Beberapa fitur yang dapat dimanfaatkan dari pustaka ini adalah manipulasi obyek HTML, penanganan event dan AJAX. Berikut ini merupakan beberapa contoh penerapan pustaka jQuery terkait fitur-fitur diatas.

MANIPULASI OBYEK HTML (DOM Traversal)

Mengakses obyek HTML melalui tag

```
1. <div>info 1</div>
2. <div>info 2</div>
3. <script src="jquery-2.0.3.js"></script>
4. <script>
5. $(document).ready(function(){
6.   $('div').html('<u>info 2</u>');
7. });
8. </script>
```

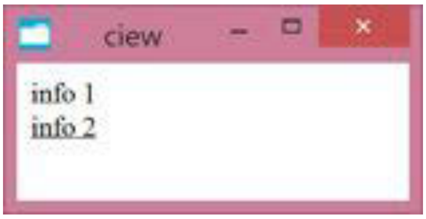
Perintah pada baris 5 dan 7 pada skrip diatas digunakan sebagai perintah jquery untuk menangani event ready dari dokumen HTML. Event ready adalah event yang dikirimkan pada saat dokumen selesai di tampilkan oleh browser.

Sedangkan perintah pada baris 6 digunakan untuk mengakses obyek HTML dengan tag div, `$('div')`, dan mengisinya dengan HTML `<ul>info 2</u>`.

Mengakses obyek HTML melalui atribut id

```
1. <div>info 1</div>
2. <div id="info2">info 2</div>
3. <script src="jquery-2.0.3.js"></script>
4. <script>
5. $(document).ready(function(){
6.   $('#info2').html('<u>info 2</u>');
7. });
8. </script>
```

Program Javascript pada contoh diatas ini mirip dengan contoh sebelumnya, bedanya pada baris 6, obyek HTML pada contoh ini diakses melalui atribud id yang ditambahkan pada tag div. Penunjukkan atribut id oleh jQuery dilakukan dengan menambahkan tanda pagar (#) diawal id dari tag yang akan diakses. Penerapan atribut id tidak hanya dapat diberikan pada tag div namun juga untuk tag lainnya. Apabila dijalankan hanya tag dengan id info2 yang akan berubah tampilannya. Seperti ditunjukkan pada gambar berikut.



Mengakses obyek HTML melalui atribut class

```
1. <div>info 1</div>
2. <div class="info2">info 2</div>
3. <script src="jquery-2.0.3.js"></script>
4. <script>
5. $(document).ready(function(){
6.   $('.info2').html('<u>info 2</u>');
```

```
});
</script>
```

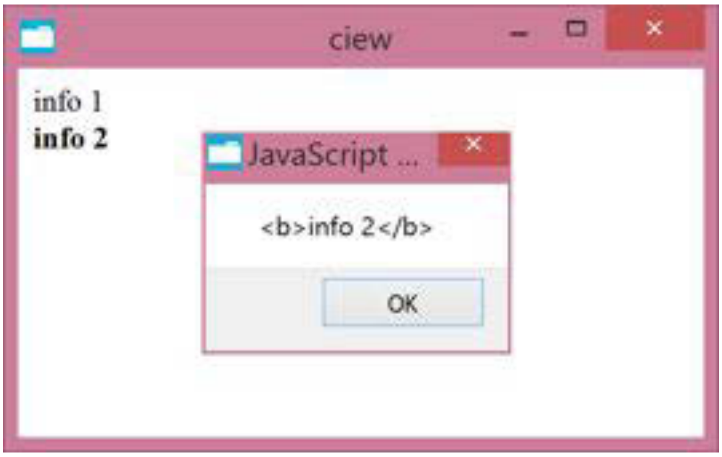
Apabila program ini dijalankan akan memberikan tampilan yang sama pada contoh sebelumnya. Kalau pada contoh sebelumnya atribut id ditunjuk dengan menggunakan tanda #, disini penunjukkan kelas dilakukan dengan menggunakan tanda titik (.) diawal nama kelasnya.

PENANGANAN EVENT

Pemberian event onclick pada obyek HTML

```
1. <div>info 1</div>
2. <div id="info2"><b>info 2</b></div>
3. <script src="jquery-2.0.3.js"></script>
4. <script>
5. $(document).ready(function(){
6.   $('#info2').click(function(){
7.         alert($('#info2').html());
8.   });
9. });
10.  </script>
```

Pada skrip ini penambahan event onclick dilakukan pada obyek dengan id info2, yakni tag div. Skripnya ditunjukkan pada baris 6-8. Apabila tag tersebut diklik maka akan ada alert yang ditampilkan. Hasilnya dapat dilihat pada gambar berikut.



c. Rangkuman.

Penambahan pustaka eksternal pada skripting halaman web dapat memiliki sejumlah nilai tambah. Selain adanya tambahan fitur pustaka eksternal seperti jQuery juga dapat memudahkan dalam memprogram HTML menggunakan Javascript. jQuery merupakan salah satu pustaka eksternal yang banyak digunakan saat ini karena bersifat terbuka dan memiliki banyak fitur dan juga pustaka turunan.

d. Tugas.

1. Buatlah program untuk melakukan perhitungan operasi aritmatika sederhana berikut ini (1 operator dan 2 operand) melalui penggunaan librari eksternal jQuery dan pustaka bawaan Javascript. Contoh hasil akhirnya sebagai berikut.





2. Buatlah program untuk pencarian array dengan kondisi berikut.
   a. Array yang digunakan adalah Jan, Feb, Mar, Apr, Mei, Jun, Jul, Agu, Sep, Okt, Nop, Des.
   b. Inputnya adalah dua bulan terpilih dari data diatas.
   c. Outputnya adalah deretan bulan dari bulan pertama dan kedua yang ditentukan pada opsi b, dengan urutan sesuai posisi bulan pada array, contohnya dengan input bulan pertama Mar dan bulan kedua Jul, maka outputnya adalah Mar, Apr, Mei, Jun, Jul.

e. Tes Formatif.

1. Apa yang dimaksud dengan pustaka eksternal?
2. Jelaskan apa saja metode yang dapat digunakan oleh pustaka jQuery dalam mengakses obyek HTML dalam suatu halaman web?

f. Lembar Jawaban Tes Formatif.

LJ- 01 :

....

....

....

....

....

....

....

....

....

....

LJ- 02 :

....

....

....

....

....

....

....

....

....

....

g. Lembar Kerja Siswa.

DAFTAR PUSTAKA

Adam J. Sontag, Raplph Withbeck, Richard D. Worth (2014), *jQuery Learning Center*, http://learn.jquery.com, diakses 17 Maret 2014, 15:24 WIB.

Arief Ramadhan, S.Kom, Student Exercise Series: Pemrograman Web dengan HTML, CSS, dan Javascript, Penerbit PT Elex Media Komputindo: Jakarta.

Betha Sidik, Ir., Husni I. Pohan, Ir., M.Eng. (Juni 2012) , *Pemrograman Web Dengan HTML Revisi Keempat,* Penerbit Informatika Bandung

Budi Raharjo (September 2011), *Belajar Pemrograman Web*, Penerbit Modula, Bandung

Jennifer Niederst Robbins (2012), *Learning Web Design 4th Edition,* Penerbit O'Reilly Media, Inc.: Kanada

Kevin Yank, Cameron Adams (2007), *Simply JavaScript*, Penerbit SitePoint Pty. Ltd.: Australia

Purnama Pupung Budi (2004), *Kiat Praktis menjadi Desainer Web Profesional*, PT Elex Media Komputindo, Jakarta

Simarmata Janner (2010), *Rekayasa Web*, Penerbit Andi, Yogyakarta

Wahana Komputer (2013), *Teknik Singkat dan Cepat Menguasai CSS3*, Penerbit Andi, Yogyakarta